

**Internal Assessment Resource**

**Digital Technologies & Hangarau Matihiko Level 1**

This resource supports assessment against Achievement Standards 91883 and 91884[[1]](#footnote-1)

**Standard title:** Develop a computer program (4 credits)

Use basic iterative processes to develop a digital outcome (6 credits)

**Credits:** 10

**Resource title:** Chatbot Charlie

**Resource reference:** Digital Technologies & Hangarau Matihiko 1.7B\_1.8B Version 1

|  |
| --- |
| This resource:   * Clarifies the requirements of the achievement standard * Supports good assessment practice * Should be subjected to the school’s usual assessment quality assurance process * Should be modified to make the context relevant to students in their school/kura environment and ensure that submitted evidence is authentic |

|  |  |
| --- | --- |
| Date version published by Ministry of Education | December 2017 Version 1  To support internal assessment from 2018 |
| Authenticity of evidence | Teachers/kaiako must manage authenticity for any assessment from a public source, because students/ākonga may have access to the assessment schedule or student exemplar material.  Using this assessment resource without modification may mean that students’ work is not authentic. The teacher may need to change figures, measurements or data sources or set a different context or topic to be investigated or a different text to read or perform. |

**Internal Assessment Resource**

**Achievement standard:** 91883 and 91884

**Standard title:**  Develop a computer program

Use basic iterative processes to develop a digital outcome

**Credits:** 10

**Resource title:** Chatbot Charlie

**Resource reference:** Digital Technologies & Hangarau Matihiko 1.7B\_1.8B Version 1

**Teacher/Kaiako guidelines**

The following guidelines are supplied to enable teachers/kaiako to carry out valid and consistent assessment using this internal assessment resource.

Teachers/kaiako need to be very familiar with the outcome being assessed by the achievement standards. The achievement criteria and the explanatory notes contain information, definitions, and requirements that are crucial when interpreting the standard and assessing students/ākonga against it.

**Context/Te Horopaki**

This activity requires students to develop a refined chatbot computer program that can hold a simple conversation by recognising words in the input sentences and providing a suitable reply.

This chatbot could be as simple as a “Magic 8 Ball” program that looks for a couple of keywords and supplies a fixed response to the input, or it could provide a choice of responses and respond to those. Any input outside of expected input will cause the chatbot to present a new question. Narrow contexts work best to begin with, however, the final program could develop more complex input analysis, basic memory, sentence reconstruction etc. This provides an almost unlimited scope to extend more able students. See the resources section for further information.

In developing their computer program, students should demonstrate the use of iterative processes to plan, trial, test and evaluate their computer program. This can be done by keeping a simple development log where they record regular reflections on their progress as they plan, develop, trial and evaluate successive iterations of their program. Students should be encouraged to regularly save versions of code as they develop towards the final outcome. Teachers could use this opportunity to teach and use version control software such as Git or CVS to help gather assessment evidence although this is not a necessity.

The goal of using an iterative process is to accurately construct a computer program, starting from the simplest first steps and improving its quality in an ongoing manner. The students will need good working knowledge of the chosen programming language before attempting this project and time should be allocated at the start of the unit to ensure this is the case. Students develop their program through the application of appropriate programming tools and techniques. The students should be encouraged to develop their own program including the questions and responses that their chatbot produces. As a result, they must ensure that they have described and addressed implications and considerations including appropriate cultural and ethical chatbot responses.

***Project based learning and collaboration***

All code should be written by individual students. Students may share ideas, questions, responses and concepts but must attempt to implement the code on their own. Any code that is not their own must be acknowledged as such and will not be used to determine their grade.

***Assessment***

You will assess students on:

* their application of appropriate tools and techniques in accurately developing their computer program
* their application of an iterative development process to improve and refine their computer program
* the extent to which their computer program has synthesised information from their planning, testing, trialling and evaluation to ensure the outcome is of a high quality
* the extent to which they have addressed relevant implications in the final computer program.

Depending on the focus of your teaching and learning programme, you may wish to include more details regarding the tools, techniques and design elements that students may demonstrate in their computer program.

Students should keep a development log to collect evidence as they complete the task. This could include written evidence, annotated screenshots, diagrams, short video clips, or code snippets.

You may want to give students guidance on appropriate style and format for their evidence log. Quality is more important than quantity. These achievement standards do not assess format or style of the evidence.

To prepare students for this assessment teachers should encourage students to examine conversations, conversation starters and the difference between open and closed questions. This will assist students to create the correct sentence sets to allow their chatbot to hold a reasonable conversation on a fixed topic.

Students need to be encouraged to choose a narrow context for their chatbot, for example a subject like sport, or Māori phrases for English words, in order to be successful. The chatbot will need to be able to greet people, but otherwise all the sentences it uses are predefined. All the bot program has to do is choose the next sentence from the list. This could be done by giving the user two options as answers. If the user chooses an option not presented, the chatbot will start a new conversation. A general chatbot that responds to a wide range of inputs, analyses the sentence structure of the input or implements learning algorithms is beyond Level 1 and therefore not expected for this assessment.

Students should be encouraged to develop reusable code that selects sentences from the sentence sets rather than code each option separately. Thus the chatbot is more of a chat engine that can easily have a conversation on a totally different topic simply by changing the sentence sets that the chatbot uses for its discussion. Students should think about the data structures and the functions they will use to achieve this.

**Iterative Design Process**

Students will be asked to keep a development log of iterations of the code. To begin with, they should decompose the program/implementation of the program into a series of increasingly complex versions to be created in each iteration. They should think about what data structures and/or what functions they will require for their chatbot program and how to develop this in sensible stages.

Each iteration should contain evidence of:

* Planning
  + A basic written plan of what the code should achieve in this iteration and how they might go about achieving it. For example, students may use pseudo code, flowcharts, diagrams or a simple written description.
  + Plan how to test this portion of the code.
* Development
  + Writing the code in the chosen language including some code comments and adhering to coding conventions.
* Trialling
  + Students should have a clear idea of what they are trialling in each iteration.
  + Students could record this information in a written paragraph, test table or annotated screenshots.
* Evaluation
  + Students should have a written paragraph, or short video clip, outlining how successful their code was at achieving its objective and their next steps.

**Suggested Starting Stages:**

1. Random response generator. Choose a response from a list of strings.
2. Looping Random Response Generator.
3. Random Response Generator as a function called from the main program.
4. Appropriate Response Generator
   1. Code looks within the user input string for a particular word (in Python using the “in” keyword) and responds appropriately. Could be a function.
5. Question/Response
   1. Focusing the scope of the conversation with a specific question like “What’s your favourite sport?” and developing code to look for particular keywords (for example “rugby”) and directing the conversation based on multiple potential responses stored in a list.
6. Student directed next steps of iterative improvement.

Each of the above stages is a potential iteration and should be recorded as a version in the student’s development log. The progressions do not stop there, however, if the students have got to this stage they could be assessed against the criteria.

**Conditions/Ngā Tikanga**

This is an individual task where students are required to code the entire program themselves. They may use all available resources including reference material, the internet and teacher or other student expertise. Students may share ideas, solutions and questions/responses but all code must be the student’s own work. Any code that is used from another source must be acknowledged and cannot be assessed as their own work.

Conditions of Assessment related to this achievement standard can be found at <http://ncea.tki.org.nz/Resources-for-Internally-Assessed-Achievement-Standards>

**Resource requirements/Ngā Rauemi**

Students will need access to hardware and software necessary to produce and test the outcome.

Links to assist:

<https://wiki.scratch.mit.edu/wiki/Artificial_Intelligence>

<https://www.youtube.com/watch?v=gZAdYyjxRcg>

<https://medium.freecodecamp.org/creating-a-chat-bot-42861e6a2acd>

<https://www.chatbots.org/ai_zone/viewthread/492/>

<https://www.codeproject.com/Articles/36106/Chatbot-Tutorial>

<https://chatbotsmagazine.com/design-framework-for-chatbots-aa27060c4ea3>

<https://apps.worldwritable.com/tutorials/chatbot/>

<https://venturebeat.com/2016/07/27/5-tips-for-writing-chatbot-scripts/>

**Additional information/He Kōrero Atu**

***Teacher Support information***

The teaching and learning of the knowledge, concepts and skills required to prepare students for this assessment, plus the assessment itself, would occupy a major part of the year’s programme. The total of 10 credits for assessment indicates that approximately 100 hours of teaching, learning (in and out of class) and assessment in a programme of study will be required – about half the year’s programme. Schedule two or more regular progress checks with the students during this activity as a minimum.

Students must be able to code in their chosen programming language before they embark on this project. They should be able to use:

* procedural, iterative and conditional statements
* lists and/or functions
* other built-in functions

**Iterative Design and Development**

***Planning, developing and testing:***

Students should plan, develop and trial use cases to test the code for every iteration. Students should provide evidence that they have tested and evaluated the code and any functions that they have written. Evidence of this testing can be recorded in their development logs.

The final outcome should include evidence that the student has recognised and addressed a range of relevant implications.

Students should develop test cases to test the code for every iteration. Students should provide evidence that they have tested and evaluated the code and any functions that they have written. Evidence of this testing can be recorded in their development logs.

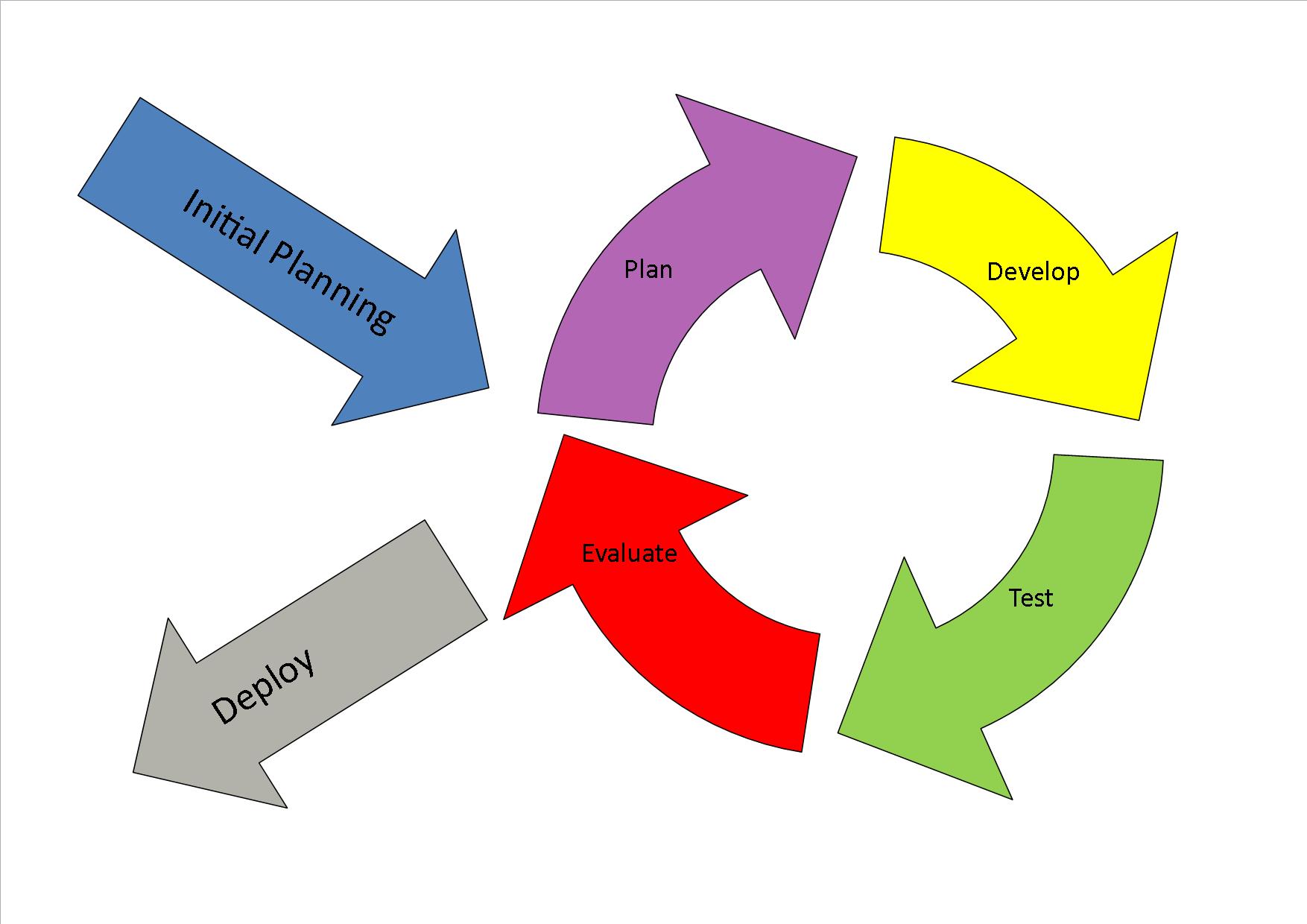
The final outcome should include evidence that the student has recognised and addressed a range of relevant implications.

**Evaluating**

Students should evaluate their project progress at each iteration, outlining what they have done so far, what they have found difficult and what the next steps should be to produce a more refined outcome. This evaluation should be based on evidence gathered during the development and testing phase, and should help students lead into any updates for their planning.

***Final Outcome:***

Students individually produce a computer program that is appropriate to the task. Teachers should ensure the rigour of the outcome is appropriate for Level 6 of the NZ Curriculum (e.g. has not been produced through simple modification of pre-designed templates). The computer program that the student is being assessed on should be original coding, which has been developed by the student. Any code that is used from another source must be acknowledged and cannot be assessed as their own work.



**Useful resources:**

Good conversation resources:

<http://www.wikihow.com/Have-a-Great-Conversation>

<http://www.improveyoursocialskills.com/conversation>

<http://techtimeout.com/10-ways-start-maintain-conversation/>

<http://examples.yourdictionary.com/examples-of-open-ended-and-closed-ended-questions.html>

<https://www.wikihow.com/Ask-Open-Ended-Questions>

<http://changingminds.org/techniques/questioning/open_closed_questions.htm>

Magic 8-ball program from <http://hawkee.com/snippet/6820/> :

Further links:

<https://wiki.scratch.mit.edu/wiki/Artificial_Intelligence>

<https://www.youtube.com/watch?v=gZAdYyjxRcg>

**Internal Assessment Resource**

**Achievement standard:** 91883 and 91884

**Standard title:**  Develop a computer program

Use basic iterative processes to develop a digital outcome

**Credits:** 10

**Resource title:** Chatbot Charlie

**Resource reference:** Digital Technologies & Hangarau Matihiko 1.7B\_1.8B Version 1

**Student/Ākonga instructions**

**Introduction/Kupu Arataki**

This assessment activity requires you to develop a computer program that simulates a basic chatbot program. It must be able to hold a simple conversation by recognising words in the input sentences and constructing a suitable reply.

You are going to be assessed on:

* how well you have applied an iterative development process to improve and refine your computer program
* how well you have used information from your planning, testing, trialling and evaluation to improve the quality of your computer program
* how well you have addressed relevant implications in your final computer program
* how well-structured, logical and flexible your final computer program is as a response to the task.

Teacher note: Insert due dates and timeframes

This is an individual task. You may share ideas, questions and concepts but must attempt to implement the code on your own. Any code that is used from another source must be acknowledged and cannot be assessed as your own work.

**Task/Hei Mahi**

You are to design a “chatbot” computer program in a text written programming language of your choice. This chatbot could be as simple as a “Magic 8 Ball” type program that looks for a couple of keywords and supplies a fixed response to the input.

In developing your computer program, you should demonstrate the use of basic iterative stages to plan, develop, test, and evaluate your computer program. This can be done by keeping a simple development log and saving versions of code as you develop the final outcome. Your development log should contain an initial plan of what you intend to make and one entry for each iteration of your program as you develop, test and evaluate to create your final outcome.

You will develop your program through the application of appropriate programming tools and techniques. Your program must demonstrate use of:

* variables storing at least two types of data (e.g. numeric, text, Boolean)
* sequence, selection and iteration control structures
* input from a user, sensors or another external source

**and one or more of:**

* data stored in collections (e.g. lists, arrays, dictionaries)
* user-defined methods, functions or procedures.

Throughout the development process you must ensure that you address any relevant implications including appropriate cultural and ethical chatbot responses.

This program will be developed in an iterative manner with multiple versions of increasingly high quality code. Each iteration should contain planning, development, testing and evaluation of the program. You will be asked to keep a development log of your progress where you will record this evidence and save versions of your code as you develop it.

**Iterative Design Process**

You are to keep a development log of iterations of the code.

To begin the project, you should plan a series of potential steps you are going to take. Planning for each iteration will be informed by your development, testing and evaluation of the previous step. You should think about what data structures and/or functions you will require for your chatbot program at each stage.

Your development log should contain evidence of the following for each iteration:

* Planning
  + A basic written plan of what the code should achieve in this iteration and how you might go about achieving it. For example, students may use pseudo code, flowcharts, diagrams or a simple written description. Ensure you have thought about how to test the code (specific test cases).
* Development
  + Include your code at this stage, including some code comments and adhere to coding conventions where they are appropriate. This can be the actual file saved as a version or a screenshot of the code for this stage.
* Testing
  + You should record this information in a written paragraph, test table, screen cast or annotated screenshots.
  + Include what you expect to happen and the actual outcome of the test.
* Evaluation
  + You should have a written paragraph or short video clip outlining how successful or unsuccessful your code was at achieving its objective. You must also outline what the next steps will be as a result of your latest development and testing. You should see an incremental improvement of your program as you develop further iterations.
  + You should describe and show how you have addressed relevant implications in your development and testing.

**Hand in:**

* Your development log including evidence of planning, developing, testing and evaluation of each iteration of your program.
* The final computer program, and the versions created as you developed it.

**Resources/Ngā Rauemi**

**Example of Chatbot conversation where Charlie talks about Sport:**

Me: Hi

Charlie: Hey right back!

Charlie: Do you like sport? Yes or No

Me: Yes

Charlie: So what is your preference, Rugby or Tennis?

Me: I like rugby

Charlie: Do you support the Crusaders or the Hurricanes?

Me: I don’t like either.

Charlie: Have you watched the All Blacks playing recently?

**Note:** Here Charlie offers the user input choices. Charlie will only respond to these choices, but has a few different possible responses to each of the input options. A random number will be used to choose the next response. Charlie will also start a new conversation if the expected answer is not received. In this way, no two conversations are exactly the same.

**Assessment schedule/Mahere Aromatawai: Digital Technologies & Hangarau Matihiko** **91883 – Chatbot Charlie**

|  |  |  |
| --- | --- | --- |
| **Evidence/Judgements for Achievement/Paetae** | **Evidence/Judgements for Achievement with Merit/Kaiaka** | **Evidence/Judgements for Achievement with Excellence/Kairangi** |
| Develop a computer program.  The student has:   * written a program that performs the specified task, using a suitable programming language   **For example (partial evidence)**  The student has written a simple, functional chatbot program in the language of their choice which includes variables of at least two data types, sequence, selection and iteration control structures.  *The chatbot works on a sample of expected cases but may have used only one function or may use only one list (array) and is not structured very well.*  The student has:   * set out the program code clearly and documented the program with comments   **For example (partial evidence)**  *Comments are present but may not be particularly descriptive or frequent. The code has made use of indention, making it clear where each new section starts.*  *e.g. # this code creates the conversation loop.*  The student has:   * tested and debugged the program to ensure that it works on a sample of expected use cases   **For example (partial evidence)**  *There is some evidence of expected cases that were used to test and debug the program to show that the program works when the user inputs data that is expected. Testing may be trial and error rather than clearly thought out.*  *The student has written functional code that works on expected input cases. Testing has been observed by the teacher but the student has provided little evidence of formal testing. The student has written about some expected case data and has tested what happens when the program is run. Such testing may be observed and recorded by the teacher, presented in table form with minimal notes, recorded using a screen cast or described using examples.*  *The examples above are indicative samples only* | Develop an informed computer program.  The student has:   * documented the program with variable names and comments that describe code function and behaviour   **For example (partial evidence)**  *There are clear comments throughout the code that help to describe relevant functions or sections of code. The variable names clearly describe the data they hold.*  *e.g. # this function selects the appropriate sentence for the chatbot to say next. It will present the next response.*  *e.g. response = pick\_next\_sentence(word\_used)*  The student has:   * followed conventions of the chosen programming language   **For example (partial evidence)**  *Python files and functions contain a docstring explaining the purpose of the program/function. Constants are ALL\_CAPS with underscores separating the words if required.*  *Variable names use underscore rather than Camel case.*  *Functions appear before the main section of code and the main section is all at bottom, not between the functions, thus making the program easier to read.*  The student has:   * tested and debugged the program in an organised way to ensure that it works on a sample of both expected and relevant boundary use cases   **For example (partial evidence)**  *The student tests frequently during development (observed and recorded) and the final program works when the user inputs data that is expected and checks or handles when the data is outside of specific thresholds. The student has not relied on trial and error approach and has decided upon expected and boundary cases that need to be tested.*  *Such testing may be observed and recorded by the teacher, presented in table form, recorded using a screen cast or described using examples.*  *In a situation where a user is asked to “choose the answer A or B”, the program checks that the input is A or B, and starts a new conversation if the user chooses neither A nor B.*  *The program might not correctly handle unexpected data and the conversation may be a little stilted.*  *The examples above are indicative samples only* | Develop a refined computer program.  The student has:   * ensured that the program is a well-structured, logical response to the task   **For example (partial evidence)**  *The student’s final program consists of multiple general purpose functions/methods so that the program flow is clear and there is no unnecessary or redundant code. Functions/methods are well named so that they are self-documenting.*  *The code is clean, concise, and easily readable. The main program may be short and consists of multiple reusable user defined functions which do most of the logic and processing.*  *The program has included either a function/method or use a collection to store data.*  Student code is laid out so that all the functions appear before main program.  **For example (partial evidence)**  *Function to select random next sentence is called random\_next\_sentence()*  *The code is clean, concise, efficient and easily readable. The main program may be short and consists of multiple reusable user defined functions which do most of the logic and processing.*  The student has:   * made the program flexible and robust   **For example (partial evidence)**  The student has created a program with a flexible structure through the use of functions/methods or data stored in collections, that cannot be easily improved upon.  The program uses actions, conditions and control structures effectively and without unnecessary repetition.  The program uses constants, variables and derived values in place of literals. Such variables appear at the top of the code where they can easily be seen and changed should those values change over time.  The program correctly handles expected, boundary and invalid values.  *They have used appropriate error handling techniques to ensure that program handles even unexpected cases.*  The student has:   * comprehensively tested and debugged the program   **For example (partial evidence)**  *The program checks the validity of input data and deals with expected, boundary and unexpected or invalid cases. All cases have been comprehensively tested including testing each condition within their code.*  *Testing has been done in a systematic way. Test cases have been well thought out and notes may have been made showing that the code works as expected for* ***all*** *use cases.*  *The examples above are indicative samples only* |

Final grades will be decided using professional judgement based on a holistic examination of the evidence provided against the criteria in the achievement standard

**Assessment schedule/Mahere Aromatawai: Digital Technologies & Hangarau Matihiko 91884 – Chatbot Charlie**

|  |  |  |
| --- | --- | --- |
| **Evidence/Judgements for Achievement/Paetae** | **Evidence/Judgements for Achievement with Merit/Kaiaka** | **Evidence/Judgements for Achievement with Excellence/Kairangi** |
| Use basic iterative processes to develop a digital outcome.  The student:   * plans a digital outcome to address a problem, need, opportunity or interest   **For example (partial evidence):**  *They research issues like basic chatbots, conversations, open and closed sentences, and plan how they are going to incorporate these.*  *They experiment with some basic techniques with small stand-alone programs to get a clear idea of how data structures and functions work.*  *An online tool has been used to plan out the development process.*  The student:   * develops the digital outcome by decomposing the problem   **For example (partial evidence):**  *They have decomposed their basic computer program into the components that need to be developed and tested such as data structures, functions, user interface.*  The student:   * trials components of the outcome in an iterative manner   **For example (partial evidence):**  *They have tested the code for a basic conversation loop. They next plan and test a set of subject specific questions and responses.*  *They test how next response will be selected for example readability and aesthetics. Each component is planned and tested in an iterative manner until the final chatbot program is produced. This is visible within their development log.*  The student:   * tests that the digital computer program functions as intended   **For example (partial evidence):**  *At each stage, the student tests the functionality of the computer program and that it can hold a realistic conversation. This is evidenced within their development log.*  *They carry out the tests with peers to ensure the chatbot works.*  The student:   * describes any implications   **For example (partial evidence):**  *They have spoken to students to determine whether the responses were socially acceptable. The student recognises that it is unethical to use copyrighted code as your own. They have recognised that the language the chatbot uses will affect the enjoyment of using the chatbot. However, the student may not have chosen the best solution to address the considerations or could have more fully addressed these considerations.*  *Evidence of this will be within the final outcome.*  *The examples above are indicative samples only* | Use basic iterative processes to develop an informed digital outcome.  The student:   * uses information from testing and trialling to improve the outcome   **For example (partial evidence):**  *Screen shots have been provided with a brief annotation that shows the improvements in chatbot conversation that were made after making changes to the program. They also provide a short video to demonstrate improved functionality after correcting a bug in the code.*  The student:   * trials multiple components and/or techniques and selects those which ensure the outcome functions as intended   **For example (partial evidence):**  *They have trialled two different techniques for choosing the next response and selects the choice that does not cause functionality issues.*    The student:   * addresses any implications   **For example (partial evidence):**  *They have addressed the fact that it is unethical to use copyrighted code through creating their own code for the chatbot.*  *They also ensure that any conversation is socially and culturally acceptable.*  *Usability and aesthetic considerations have been addressed through testing how the next response is presented to the user and thinking about different ways this information could be presented.*  *The examples above are indicative samples only* | Use basic iterative processes to develop a refined digital outcome.  The student:   * applies information from the planning, testing and trialling of components to develop a high-quality outcome   **For example (partial evidence):**  *Evidence has been provided that their planning, testing and trialling has allowed them to develop all components and information to a high quality.*  *Their outcome functions as intended and has no obvious errors in functionality or presentation of the information.*  *Evidence gained from trialling is thorough and organised testing has been integrated into the outcome in an ongoing manner to ensure the outcome is of high quality, including aesthetics, functionality and usability.*  *The student has clearly addressed any ethical issues that arose in the responses of the chatbot and has taken care that the chatbot conversation will not offend any user.*  *The examples above are indicative samples only* |

Final grades will be decided using professional judgement based on a holistic examination of the evidence provided against the criteria in the Achievement Standard

1. These achievement standards are derived from both *The New Zealand Curriculum* and *Te* *Marautanga o Aotearoa.* [↑](#footnote-ref-1)