**Level 3 Database Assessment**

| **Achieved** | **Merit** | **Excellence** |
| --- | --- | --- |
| **91902:** Use complex techniques to develop a database **(4 credits)** | | |
| Use complex techniques to develop a database. | Use complex techniques to develop an informed database. | Use complex techniques to develop a refined database. |

**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 standard. 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**

The assessment activity requires students to use complex techniques to create a refined relational database with a web interface. The outcome should include a search feature which allows users to explore the data. It should give authorised users the ability to log in and then add, edit and delete items in the database.

Teachers are encouraged to edit this task to make it suitable for their students and community. Please see the supporting documentation for datasets that students can use to complete this assessment. Further datasets can be found at <https://www.kaggle.com/>

**Conditions/Ngā Tikanga**

It is recommended that students should have at least two identified checkpoints with their teacher as they work through this assessment activity to ensure they have an opportunity to ask questions and gather feedback.

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 the web, digital devices and a database hosting environment (e.g. XAMPP and local host).

**Level 3 Database Assessment**

| **Achieved** | **Merit** | **Excellence** |
| --- | --- | --- |
| **91902:** Use complex techniques to develop a database **(4 credits)** | | |
| Use complex techniques to develop a database. | Use complex techniques to develop an informed database. | Use complex techniques to develop a refined database. |

**Introduction/Kupu Arataki**

The assessment activity requires you to use complex techniques to create a refined relational database with a web interface. The outcome should include a search feature which allows users to explore the data. It should give authorised users the ability to log in and then add, edit and delete items in the database

**Task/Hei Mahi**

Design the structure of the data and create a database which allows students and teachers to explore (and edit) the data.

Either choose a dataset from the material provided in the support files area or source an alternate dataset from <https://www.kaggle.com/>. Note that if you choose to source your own data, you **must** get teacher approval to ensure that the data you have chosen will allow you to achieve this standard.

The **database** must include the following functionality:

* Users with admin access should be able to add, delete or edit entries in the database
* The database needs to be easy to search based on key fields

The **interface** must:

* Ensure that admins can’t add obviously incorrect data (perform some data validation).
* Allow users to search the database for:
  + Specific items (by name/part of a name)
  + Category (where results are sorted alphabetically)
* Allow admins to:
  + Add, edit and delete items from the database
  + search for items so that it is easy to find data that needs to be changed / removed.
* Present the data appropriately for the task and the end users.

**Documentation Requirements**

**Planning**

* a description of the table/s and data types you plan on using used
* a diagram showing how the tables are linked together (i.e. a UML diagram or similar), and a second diagram showing how the pages of the site/database are related
* a list of queries/features to be implemented with test data for each query (i.e. how will you ensure that the query/feature works correctly?).

**Relevant Implications**

It should be clear from your outcome that the relevant implications have been addressed. If you have implemented specific features to address an implication, you should include a purple slide discussing how you have addressed the relevant implication concerned.

**Testing**

Please provide evidence that your outcome has been comprehensively tested. This could include in the form of screencasts / screen shots that show…

* that the data entry form works for expected input.
* the error messages that appear when users attempt to enter invalid data.
* that the various queries work as expected.
* that the outcome was tested with end user/s to confirm it meets the purpose and end user considerations.

**Improvement of the database**

* You should improve the database through feedback and cycles of trialling and testing iteratively throughout the design, development and testing process to improve the quality of the database.

| **Submission**…  Your submission should include…   * A link to your working database * A link to the code on github * A slide show / document which includes evidence of planning, addressing relevant implications (where necessary) and testing. You have been supplied with a template to help you do this successfully. |
| --- |

**Assessment 91902 – Complex Database**

| **Evidence/Judgements for Achievement/Paetae** | **Evidence/Judgements for Achievement with Merit/Kaiaka** | **Evidence/Judgements for Achievement with Excellence/Kairangi** |
| --- | --- | --- |
| Use complex techniques to develop a database.  The student has:   * designed the structure of the database * used appropriate tools and techniques to organise, query and present data for a purpose and end users   **For example (partial evidence)**  They have:   * a database plan which includes at least two linked tables, shows the structure of the tables, shows the designs for the interfaces and a plan for linking the tables * used appropriate data types and data formatting * written a custom query to filter and/or sort data * created a form which allows users to add data to the database via the website * allowed authorised users to add, edit and delete information in the database.   The student has:   * applied appropriate data integrity and testing procedures   **For example (partial evidence)**   * data is validated before it is added to the database to prevent obviously invalid material from being inserted (e.g. blank records) * evidence of testing has been supplied showing that adding material to the database works correctly and that the various queries return the expected results * a password is required to access the admin area – if the incorrect password/username is entered, access to that area is denied (i.e. database has appropriate permissions).   The student has:   * addressed the relevant implications   **For example (partial evidence)**  The student’s outcome:   * Works as expected (functionality) * Is easy to use / has clear error messages (usability) * Looks attractive (aesthetics) * If images have been used, the source is acknowledged / they are from the public domain (IP / Copyright) * The source of the data is acknowledged (IP / copyright).   *The examples above are indicative samples only* | Use complex techniques to develop an informed database.  The student has:   * used information from testing procedures to improve the quality of the database * structured, organised and queried the data logically   **For example (partial evidence):**  The student has attempted to remove redundant data from the database.  The student has asked a volunteer to test the website/database to ensure that they can search for the items, enter and correct their data. They have made improvements/changes based on this testing.  The student has tested the website/database with the Admin person to ensure they can log on and delete/edit database entries.  The database output is displayed in a logical order that is easy to read and understand.  The queries make it easy to find data both for front end users and for admins wishing to make changes to the data via the admin panel  Students supply evidence demonstrating how they logically structured, organised and queried the data.  *The examples above are indicative samples only* | Use complex techniques to develop a refined database.  The student has:   * evidence of iterative improvement throughout the design, development and testing process   **For example (partial evidence):**  They improved the database structure and design following feedback on trial designs.  They used results from trialling and testing to improve the functionality and reliability of the database.  For instance:   * they modified the input form to make it easier to enter data after testing showed that users made errors * they improved the query results after trialling and testing to make them easier to read and understand * after testing showed some problems, they made it easier for users and admins to easily find and use the search bar * they modified the structure of the tables to make the most common queries run more efficiently.   The database has minimal redundant data.  The student has:   * used efficient tools and techniques in the outcome’s production   **Fo**r **example (partial evidence):**  They planned the structure of the database and wire framed the interface before developing the outcome. They applied version control methods. Their testing was planned and systematic, not trial and error.  The student has used “includes” to make efficient use of PHP scripts.  The student created relationship diagrams before implementing the table structure.  The student created queries that use input parameters so that scripts do not have to be duplicated.  The student has:   * presented data effectively for the purpose and end users   **For example (partial evidence):**  They ensured that the data was presented in such a way that it was able to be effectively analysed and searched (e.g. drop-down lists where appropriate, check boxes) and easily understood. There are very few grammatical or typographical errors. The layout demonstrates effective application of design principles.  *The examples above are indicative samples only* |