|  |  |
| --- | --- |
| **Task** | **File** |
| Part 1 / Task 1 | makeabooking.html |
| Part 2 / Task 1  Part 2 / Task 2 | motueka.sql |
| Part 3 / Task 1a | currentbookings.php |
| Part 3 / Task 1b | bookingdetailsview.php |
| Part 3 / Task 2 | bookingpreviewbeforedeletion.php |
| Part 3 / Task 3 | editabooking.php |
| Part 4 / Task 1  Part 4 / Task 2 | makeabooking.php  roomsearch.php |
| Part 5 / Task 1 | addusersencryptedpasswords.php  login.php |

Part 2 / Task 3)

Encryption, digital certificates and passwords can be used to protect parts of Motueka B&B’s web application.

Through its web application, Motueka B&B collects and stores data from customers such as email addresses and phone numbers. In the future, the web application may collect and store payment information as well. It is important to collect and store this data securely against data and privacy breaches as unauthorised access could lead to financial risk (for example, through stealing customers’ credit card details) or theft risk (for example, booking information could lead a thief to know when a customer will be away from their residence).

Examples of the web application collecting data include when a new customer registers and when a new booking is created. Examples of the web application storing data include storing data of its customers and bookings that can then be viewed in their ‘view customers’ and ‘view bookings’ pages.

Encryption refers to turning readable data into scrambled code such that only authorised users can access it.  Data can be encrypted when it is collected and when it is stored.  To encrypt data upon collection, a digital certificate and a key need to be configured so that the web application uses HTTPS. (“Keeping business data safe with encryption”, CertNZ)

Passwords can help prevent unauthorised access to the Motueka SQL database and to the web application as a whole.  In the web application, the ‘addusersencryptedpasswords.php’ file adds three users to the customer table (admin, ordinary customer, non customer) using the ‘password\_hash’ function to encrypt the password using PHP. (“Encrypting passwords using PHP”) The login page then utilises the ‘password\_verify’ function with the plain text and encrypted passwords. (“Creating a login page”)

Source:

https://www.cert.govt.nz/business/guides/keeping-business-data-safe-with-encryption/