

Untuk menambahkan log di edit.php guna memastikan bahwa data diupdate dan diterima dengan benar, kita dapat mengikuti pendekatan yang sama seperti di add.php. Berikut adalah cara untuk melakukannya:

1. edit.php

php

Copy code

```
<?php
```

```
/**
```

```
 * Edit an existing item
```

```
 *
```

```
 * This script handles the editing of an existing item by receiving user input,
```

```
 * validating it, and updating the item in the database. It also logs the input
```

```
 * and update status to a log file.
```

```
 */
```

```
// Include the authentication and database connection files
```

```
include("auth.php");
```

```
include("db.php");
```

```
// Check if the user is logged in
```

```
check_login();
```

```
// Get the item ID from the URL
```

```
$id = $_GET['id'];
```

```
// Log file
```

```
$log_file = "edit_log.txt";
```

```
/**
```

* Log message to a file

*

* This function writes a message to the specified log file with a timestamp.

*

* @param string \$message The message to log

* @param string \$file The log file path

*/

```
function log_message($message, $file) {
```

```
    // Get the current timestamp
```

```
    $timestamp = date("Y-m-d H:i:s");
```

```
    // Format the log message
```

```
    $log_message = "[" . $timestamp . " ] " . $message . "\n";
```

```
    // Write the log message to the file
```

```
    file_put_contents($file, $log_message, FILE_APPEND);
```

```
}
```

```
// Check if the form is submitted using POST method
```

```
if ($_SERVER['REQUEST_METHOD'] == "POST") {
```

```
    // Get the submitted name and description
```

```
    $name = $_POST['name'];
```

```
    $description = $_POST['description'];
```

```
    // Log the received input
```

```
    log_message("Received input for update - ID: $id, Name: $name, Description: $description", $log_file);
```

```
    // Validate the input
```

```
    if (!empty($name) && !empty($description)) {
```

```

// Create an SQL query to update the item in the database

$query = "UPDATE items SET name='$name', description='$description' WHERE id=$id";

// Execute the query
if (mysqli_query($conn, $query)) {
    // Log the successful update
    log_message("Successfully updated - ID: $id, Name: $name, Description: $description", $log_file);

    // Redirect to the main page after successful update
    header("Location: index.php");
    die;
} else {
    // Log the update error
    log_message("Failed to update - ID: $id, Error: " . mysqli_error($conn), $log_file);

    // Display an error message
    echo "Error: " . mysqli_error($conn);
}
} else {
    // Log the invalid input
    log_message("Invalid input for update - ID: $id, Name: $name, Description: $description", $log_file);

    // Display an error message if the input is not valid
    echo "Please enter valid information!";
}
} else {
    // Create an SQL query to retrieve the item data from the database
    $query = "SELECT * FROM items WHERE id=$id LIMIT 1";

```

```

// Execute the query
$result = mysqli_query($conn, $query);

// Fetch the item data
$row = mysqli_fetch_assoc($result);
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>Edit Item</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
    <form method="post">
        <div>Edit Item</div>
        <input type="text" name="name" value="<?php echo $row['name']; ?>"><br><br>
        <textarea name="description"><?php echo $row['description']; ?></textarea><br><br>
        <input type="submit" value="Save"><br><br>
    </form>
</body>
</html>

```

Penjelasan Kode

1. Log File Configuration:

- Menentukan file log (edit_log.txt) untuk menyimpan pesan log.

2. Log Function:

- log_message(\$message, \$file) adalah fungsi yang menulis pesan log ke file dengan timestamp.
- date("Y-m-d H:i:s") digunakan untuk mendapatkan timestamp saat ini.

- `file_put_contents($file, $log_message, FILE_APPEND)` menambahkan pesan log ke file tanpa menghapus log sebelumnya.

3. Logging Received Input:

- Setelah menerima input dari pengguna (`$name` dan `$description`), kita mencatat input tersebut ke file log dengan `log_message("Received input for update - ID: $id, Name: $name, Description: $description", $log_file)`.

4. Validation and Update:

- Jika input valid, kita membuat query SQL untuk memperbarui item di database.
- Jika query berhasil dieksekusi, kita mencatat keberhasilan tersebut ke file log dengan `log_message("Successfully updated - ID: $id, Name: $name, Description: $description", $log_file)`.
- Jika query gagal, kita mencatat kesalahan tersebut ke file log dengan `log_message("Failed to update - ID: $id, Error: " . mysqli_error($conn), $log_file)` dan menampilkan pesan kesalahan.
- Jika input tidak valid, kita mencatat input yang tidak valid tersebut ke file log dengan `log_message("Invalid input for update - ID: $id, Name: $name, Description: $description", $log_file)` dan menampilkan pesan kesalahan.

Dengan menambahkan kode ini, setiap kali ada upaya untuk mengedit item, proses tersebut akan dicatat di file `edit_log.txt`. Ini akan membantu dalam debugging dan memastikan bahwa semua input dan operasi update berhasil atau mencatat kesalahan jika terjadi.