Bytexl’s guided project

Final Project report

|  |  |
| --- | --- |
| Name of the educator | Aarshi Gupta |
| Project title | Hotel management |
| Tools / platforms used | PHP visual studio git hub xaamp server |

**About the Project (Max 250 Words)**

The project is a web-based CRUD (Create, Read, Update, Delete) application built using PHP. It allows users to manage records within a database, including adding new entries, updating existing ones, searching for records, and deleting unwanted entries. The application serves as an efficient tool for managing large datasets, especially in contexts like student management systems, employee directories, or inventory tracking. The design emphasizes simplicity, ease of use, and scalability, ensuring users can interact with the database effortlessly. It leverages a MySQL database for data storage and includes essential user authentication for secure access. The system also includes error handling and validation mechanisms to ensure data integrity and smooth user interaction.

**System Requirements**

**Hardware:**

* Processor: 1 GHz or faster
* RAM: Minimum 2 GB
* Storage: 50 MB free space

**Software:**

* Web server: Apache 2.4 or higher
* PHP: 7.4 or higher
* Database: MySQL 5.7 or higher
* Browser: Chrome, Firefox, or Safari for testing
* Git: For version control

**Functional Requirements**

* User login and authentication
* CRUD operations (Create, Read, Update, Delete)
* Data validation and error handling
* Pagination of records
* Search functionality for records
* Responsive design for user accessibility
* Integration with version control (GitHub)

**User Interface Requirements**

* Simple, responsive interface with form validation
* Intuitive layout for easy navigation
* Visual feedback on actions (e.g., record added, updated, or deleted)
* Mobile and desktop compatibility

**Inputs and Outputs**

**Inputs:**

* User credentials (username, password)
* Data for new records (name, age, email, etc.)
* Search queries for filtering records

**Outputs:**

* Successful login and user redirection
* Display of records in a table with pagination
* Success/error messages for actions (add, edit, delete)

**List of Subsystems**

1. **User Authentication:** Handles login and security.
2. **CRUD Operations:** Manages record creation, retrieval, updates, and deletions.
3. **Search & Pagination:** Facilitates record search and pagination.
4. **Error Handling & Validation:** Ensures data integrity and smooth operation.
5. **Database Integration:** Manages communication with the MySQL database.

**Other Applications Relevant to Your Project**

This project can be applied in various contexts, including:

* **Educational Systems:** Student record management systems.
* **Inventory Management:** Tracking and updating product inventory.
* **Employee Management:** Managing employee details like attendance, payroll, and performance.
* **CRM Systems:** Storing and updating client data for customer relationship management.

**Designing of Test Cases**

1. **User Login Test Case:**
   * **Function:** Validates user authentication and login functionality.
   * **Expected Outcome:** Successful login or error message for invalid credentials.
2. **Add Record Test Case:**
   * **Function:** Verifies that records can be successfully added to the system.
   * **Expected Outcome:** New record is added and displayed.
3. **Update Record Test Case:**
   * **Function:** Checks the ability to update existing records.
   * **Expected Outcome:** Record is updated in the database and displayed correctly.
4. **Delete Record Test Case:**
   * **Function:** Ensures records can be deleted.
   * **Expected Outcome:** Record is removed and no longer visible.
5. **Search Function Test Case:**
   * **Function:** Verifies the search functionality.
   * **Expected Outcome:** Correct records are displayed based on the search query.

**Future Work**

* **Mobile App Integration:** Expand the project by creating a mobile version of the application.
* **Data Analytics:** Implement reporting and analytics features for better insights into the records.
* **Security Enhancements:** Add role-based access control (RBAC) to manage user permissions.

**References**

* PHP Documentation: <https://www.php.net/>
* MySQL Documentation: <https://dev.mysql.com/doc/>
* W3Schools PHP Tutorial: https://www.w3schools.com/php/

**Reflection of the Project Creation**

1. **Technical Challenges:**
   * Integrating the backend logic with the frontend interface posed challenges, especially when dealing with pagination and dynamic data rendering. Ensuring smooth communication between the PHP code and MySQL database required attention to error handling and data validation.
2. **Software Engineering Techniques:**
   * Using the MVC (Model-View-Controller) design pattern helped separate the concerns of the application and improve maintainability. Implementing unit tests ensured that each component worked independently, making debugging easier.
3. **Benefits Experienced:**
   * I gained deeper insight into web development practices, particularly around PHP and MySQL. It also helped strengthen my problem-solving skills, especially in handling user input and errors in web applications.
4. **Other Knowledge Needed:**
   * Knowledge of advanced JavaScript for enhanced front-end interactivity and understanding of APIs for future integration with third-party services would have been beneficial.