Project Documentation

---

1. **Project Title**
   "Activity List database system"

**2. Project Description**

This application serves as an activity list for the user, it allows creating, viewing, updating, and deleting activities each detail like title descriptions past or current action users activity, log and time track,

**3. Objectives**

To offer CRUD system to users that will allow them to manage and track their activities

For the team to apply the CRUD operations in web developments and to interact with MySQL database using PHP

Develop a simple but functional PHP database/ PHP project

**4. Tools and Technologies**

HTML5, CSS3, PHP, MySQL, XAMPP (Apache and MySQL)

**5. Features** Example:

- Create: Add new activities to the database.
- Read: Display all activities from the database.
- Update: Modify existing activities data.
- Delete: Remove all data from the user database.

**6. Folder Structure**
Example:

```
crud_app/
├── db.php         (Database connection file)
├── index.php      (Main page showing users list)
├── add.php        (Form to add a new user)
├── edit.php       (Form to edit user data)
├── delete.php     (Handles deletion of a user)
├── style.css      (CSS file for styling)
└── crud_db.sql    (SQL file to create database and table)
```

**7. Database Structure** Example:

Database Name: crud_db

Table Name: users

Fields:
- id: INT, Primary Key, Auto Increment
- name: VARCHAR(100)
- activity: VARCHAR(100)
- timestamp: TIMESTAMP(100)
- status: VARCHAR(100)
- category: VARCHAR(100)

SQL Script:

```
CREATE DATABASE IF NOT EXISTS crud_db;
USE crud_db;

CREATE TABLE IF NOT EXISTS users (     id
INT(11) NOT NULL AUTO_INCREMENT,
name VARCHAR(100) NOT NULL,
activity VARCHAR(100) NOT NULL,
timestamp TIMESTAMP(100) NOT NULL,
status: VARCHAR(100) NOT NULL,
category: VARCHAR(100) NOT NULL,
PRIMARY KEY (id)
);
```

**8. Installation Steps**

1. Install XAMPP and run Apache & MySQL.
2. Copy 'crud_app' folder to 'htdocs' directory in XAMPP.
3. Open phpMyAdmin at http://localhost/phpmyadmin.
4. Import crud_db.sql to create the database.
5. Run the project in browser at http://localhost/crud_app/index.php.

**9. Usage Guide**

-Navigate to the main page to view all users with the activities
-Click “Add user/activity” button to create new record
-Use the “Edit” button to modify user details
-Use the “Delete” button remove a user along with its data

**10. Future Enhancements**

- Add activities input validation.
- Use prepared statements to prevent SQL injection.
- Improve UI using a CSS framework like Bootstrap.
- Add AJAX for asynchronous form submission.- Implement pagination for large datasets.

**11. Author**

Closas, Josiah Orchy
Piedad, Kyle Godwin
Rebucas, Brix Louie
Turtosa, John William

Date Created: 2025-4-30

**12. Physical Data Model**

The physical data model represents how the database is structured. Below is the structure of the 'users' table used in the CRUD application.

| Field Name | Data Type | Length | Description |
|---|---|---|---|
| id | INT | 11 | Primary key, auto-incremented ID |
| Full Name | VARCHAR | 100 | User's full name |
| Activity | VARCHAR | 100 | Activities name |
| Timestamp | TIMESTAMP | 100 | User time &date of the activity |
| Status | VARCHAR | 100 | Activity Status |
| category | VARCHAR | 100 | Which category the activity belongs |

**Rubrics**

The evaluation criteria for the Project. Each criterion is rated on a scale of 1 to 5, where 5 is Excellent and 1 is Poor.

| Criteria | Excellent (5) | Good (3) | Satisfactory (2) | Needs Improvement (1) | Score |
|---|---|---|---|---|---|
| Functionality | All CRUD operations work flawlessly | Most features work properly | Basic operations work | Many features are broken or missing | |
| User Interface | Clean, responsive and user-friendly | Functional and mostly responsive | Usable but not styled well | Unattractive or confusing UI | |
| Code Quality | Wellstructured, clean and commented | Mostly clean with some comments | Readable but lacks structure | Messy and hard to follow | |

| | | | | | |
|---|---|---|---|---|---|
| Database Design | Efficient structure with proper types | Appropriate use of data types | Basic but functional structure | Poorly designed schema | |
| Documentation | Complete and professional | Mostly complete | Somewhat complete | Incomplete or missing | |