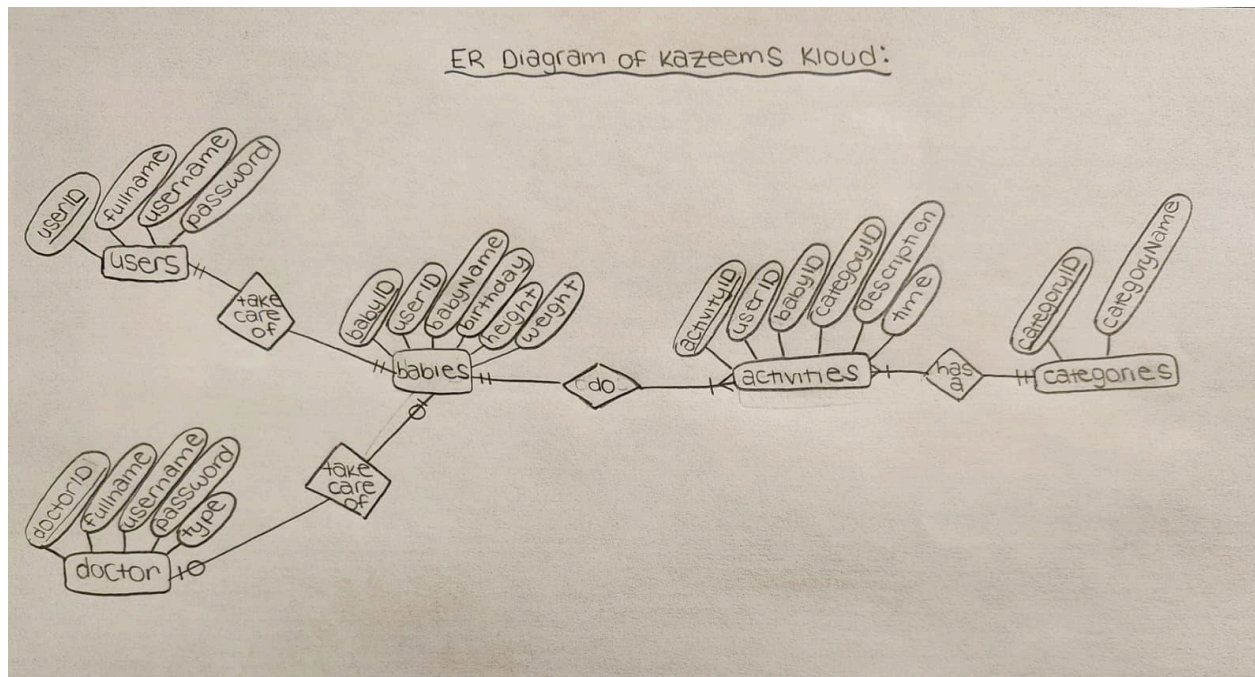


Summary, Domain, and Purpose:

A summary of my project, “Kazeem’s Kloud” is that I wanted to create a baby-tracking website. The website allows the user (mother, father, guardian) to track everything that has to do with the child. They can start by creating an account, then, they track things like what the baby eats or drinks (solids vs baby food or any medicine consumed, formula vs breastmilk, and quantity), how often the diaper was changed, what activities they do (playtime), and how often the baby goes down for a nap. Basically, this works to help the user keep track of things that they may otherwise forget. The results are combined into a weekly or monthly format so it can be seen on a larger scale, which the child’s doctor also has access to.

This website is very personal to me because it would solve a problem we are currently having with my 7-month-old nephew, Kazeem (hence the name, Kazeem’s Kloud (as in Cloud)). The website idea comes from whenever Kazeem is at my house, and we all take ‘shifts’ taking care of him. A frequent problem I have run into time and time again was that I would get Kazeem, and the previous person would go to sleep or their job or simply just forget to give me an update on what time he was fed, diaper changed, and slept. So, my solution to this problem is to create an application that allows the person in charge of Kazeem to record whatever they do with Kazeem in real time as they do it. This way, the next person simply needs to look at the application and they can see exactly what time he was last fed, burped, etc. Another use for the application is to see the progression he has made over time. When daily logs are created by the user, weekly and monthly reports can be created using that data, which allows the guardian to see the changes on a bigger scale and makes them more relative, as it is hard to notice changes on a daily basis. These reports can then also be useful for doctors and pediatricians so they get insight into the lifestyle of the baby during their appointments, allowing for a better check-up.

ERD:



Explanation:

The entities in this ERD are: Users, Doctor, Babies, Activities, and Categories.

The attributes in each entity are:

User

- userID (PK)
- fullname
- username
- password

Doctor

- doctorID (PK)
- fullname
- username
- Password
- type

Babies

- babyID (PK)
- userID (FK)
- babyName
- birthday
- height
- weight

Activities

- activityID (PK)
- userID (FK)
- babyID (FK)
- categoryID
- description
- time

Categories

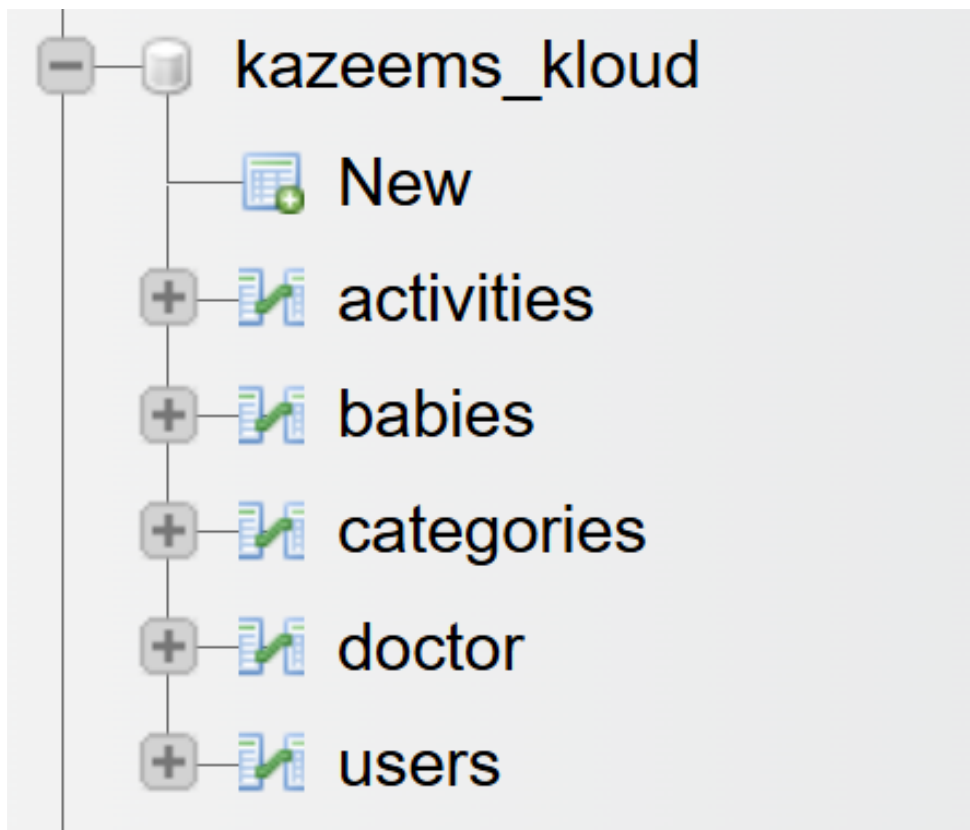
- categoryID (PK)
- categoryName

There is a ‘mandatory one’ relationship on both sides between the User and the Baby since it is mandatory that there be at least one user for each baby, and each user has at least one baby to use the application. There is an ‘optional one’ relationship between the doctor and the baby since it is not mandatory for the doctor to use this application to view the baby’s records, making it an optional aspect for both the doctor and baby in this relationships. There is a ‘mandatory many’ relationship between the baby and activities, as the baby must have multiple numbers of feedings, diaper changes, naps, and playtime in a day, week, or month. There is a ‘mandatory one’ relationship between the activities and categories since the activity must fall into one category but there can be multiple activities within the same category.

Developing & Testing:

Problems I faced when making Kazeem's Kloud website:

1. **Problem #1:** The first issue I ran into was with my database. I originally planned on splitting up each activity into its own table. Originally, I thought this would be easier since everything was separate; however, once I started creating the website, I ended up changing this to have all activities in one table and split them up by category. I found that using one activity table was easier to keep track of and kept the code neater.



2. **Problem #2:** The second issue I ran into was an issue I caused myself. I started mixing up variable names like category_id and categoryID or activity_id and activityID. This created an issue that was not easy for me to solve because I didn't realize this small mistake. After very carefully reviewing my code, I was able to pinpoint the issue and then change the variable names so they matched where they should.

3. **Problem #3:** This problem was a problem that existed because of problem #2. Since I had the wrong variable names (user_id vs userID), some of my variables weren't being passed correctly. This issue resolved itself once I fixed the variable naming throughout the website code.
4. **Problem #4:** When I first made the baby dropdown, it had the babyID not the name. Since the user might not automatically know which ID number is associated with which baby, I changed this so that the dropdown shows the baby's names, making it easy for the user to select right child.

Welcome to Kazeems Kloud

We keep track of activities so you dont have to!

Select Baby:

Kazeem	▼
Aayan	
Kazeem	

5. **Problem #5:** An issue I ran into creating this website, was that I wasn't sure how to create the reports page. After searching online, I found help to create the type of page in the style I was looking for. At first, I was going to split it up by category, but then decided to just show the number of activities in each category and then one large list of them. The code for this part was tricky for me.

Reports

Report Type:

Weekly

Date: 11/24/2025



Generate Report

Week Summary

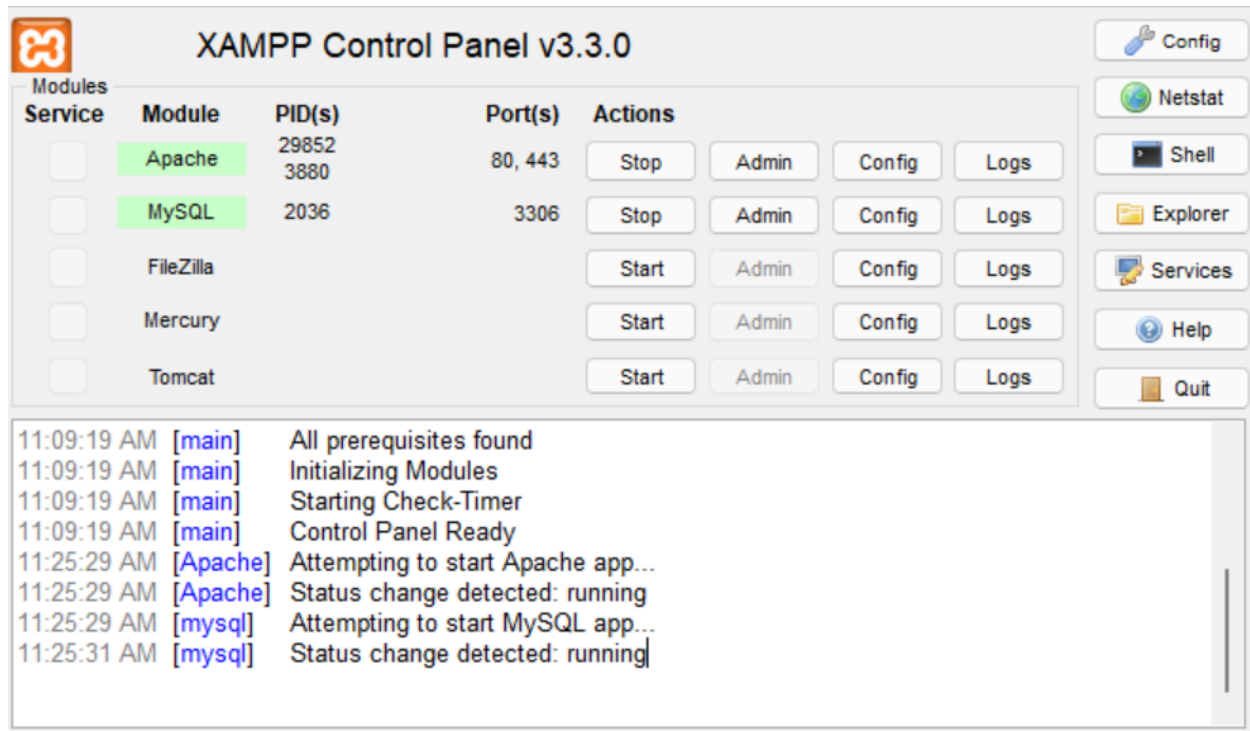
Nov 24, 2025 – Nov 30, 2025

Category	Total Activities
Diaper Changes	2
Feedings	2
Naps	1
Play Time	2

Activity Details

Date & Time	Category	Description
Nov 24, 2025 8:24 PM	Feedings	had 7 oz of formula

6. **Problem #6:** I ran into an issue in the middle of creating the Kazeems Kloud website, when I tried to launch the XAMPP Control Panel and start 'MySQL', it would not start. I was able to resolve this using the document you provided in d2l and had to get rid of my current data folder and use the backup. Once I changed the folder, 'MySQL' started back up.



7. **Problem \$7:** When I tried to add an activity, the date would be correct, but the time was always wrong. I found out it was because of the timezone so I added this line of code in the addActivities page:

```
date_default_timezone_set('America/New_York');
```

This set the timezone to EST so the time recorded was accurate.

```

addActivity.php  addActivityForm.php  deleteActivity.php  login.php  register.php  report.php
1  <?php
2  //sets the time zone to EST
3  date_default_timezone_set('America/New_York');
4
  
```

8. At first, I didn't add any way to create an account, which I realized was an issue since accounts had to be made. I ended up adding two options, one for users and one for doctors. I created the register, registerDoctor.php and registerDoctorForm.php for this.

You are connected to the database.

Username:

Password:

Login

New user? [Create an account](#)

Are you a doctor? [Register here](#)

PHP Crud:

CRUD (Create, Read, Update, and Delete) are all handled in Kazeems Kloud website:

C- Create: The user has the option to create an activity. They can have the option to add an activity to each category.

R - Read: The user can view all of the activities by clicking on the reports link. They can also see the activities on the homepage to read through any of them

U - Update: The user has the option to update any of the activities, they just have to click on the update button.

D - Delete: The user can delete any of the activities, they just have to click on the delete button.

C-Create: The query below allows the user to create a new activity in the database. This is from addActivity.php:

```
$query = 'INSERT INTO activities
        (categoryID, babyID, userID, time, description)
        VALUES
        (:categoryID, :babyID, :userID, :time, :description)';

$stmt = $db->prepare($query);
$stmt->bindValue(':categoryID', $categoryID);
$stmt->bindValue(':babyID', $babyID);
$stmt->bindValue(':userID', $userID);
$stmt->bindValue(':time', $time);
$stmt->bindValue(':description', $description);
$stmt->execute();
```

This is the code for creating a user account. This is from register.php:

```
INSERT INTO users (username, passwordHash, fullName)
VALUES (:username, :passwordHash, :fullName);
```

This is the code for creating a doctor account. This is from registerDoctorForm.php:

```
INSERT INTO doctor (username, passwordHash, fullName, type)
VALUES (:username, :passwordHash, :fullName, :type);
```

R-Read: This query lets the user read or view all the activities in the database from that specific user. This is from index.php:

```
$queryProducts = 'SELECT a.*, b.babyName
FROM activities a
JOIN babies b ON a.babyID = b.babyID
WHERE a.userID = :user_id
AND a.babyID = :baby_id
AND a.categoryID = :category_id
ORDER BY a.activityID';
```

This code/query lets the user see the baby. This is from index.php:

```
$query = 'SELECT babyID
FROM babies
WHERE userID = :user_id
LIMIT 1';
```

This code/query lets the user see the categories. This is from index.php:

```
$query = 'SELECT *
FROM categories
ORDER BY categoryID';
```

U-Update: This query lets the user update of the previous activities, if they need to make any edits or changes. This code is from updateActivities.php:

```
$query = 'UPDATE activities
SET categoryID = :categoryID, babyID = :babyID,
userID = :userID, time = :time, description = :description
WHERE activityID = :activityID';
```

D-Delete: This query lets the user delete any of the activities from the database at any given time ,it is from deleteActivity.php:

```
$query = 'DELETE FROM activities
        WHERE activityID = :activityID';
$stmt = $db->prepare($query);
$stmt->bindValue(':activityID', $activityID);
$stmt->execute();
```

Session handling: At the top of every file there is a session.start() line to initialize a session.

Security: The security measures in place right now are hashing the users password so they are not visible in the database. Also new sessions are started on every page.

The forms all work without errors. The forms send data using POST and checks all the attributes. Some examples:

updateActivities.php:

Data Handling/Receiving the values using POST:

```
$activityID = filter_input(INPUT_POST, 'activityID', FILTER_VALIDATE_INT);
$categoryID = filter_input(INPUT_POST, 'categoryID', FILTER_VALIDATE_INT);
$babyID = filter_input(INPUT_POST, 'babyID', FILTER_VALIDATE_INT);
$userID = filter_input(INPUT_POST, 'userID', FILTER_VALIDATE_INT);
$time = filter_input(INPUT_POST, 'time');
$description = filter_input(INPUT_POST, 'description', FILTER_SANITIZE_STRING);
```

Validating the inputs:

```
if ($activityID === null || $activityID === false ||
    $babyID === null || $babyID === false ||
    $userID === null || $userID === false ||
    $time === null || $time === false) {
    //error message if anything is wrong
```

```
$error = "Invalid product data. Check all fields and try again.";
echo $error;
}
```

updateActivitiesForm.php:

Validating:

```
if($activityID == null || $activityID === false){
    $error = "Error";
    echo $error;
    exit();}
```

addActivity.php:

Data Handling/Fetching inputs using POST:

```
$categoryID = filter_input(INPUT_POST, 'categoryID',
    FILTER_VALIDATE_INT);
$babyID = filter_input(INPUT_POST, 'babyID', FILTER_VALIDATE_INT);
$userID = filter_input(INPUT_POST, 'userID', FILTER_VALIDATE_INT);
$description = filter_input(INPUT_POST, 'description', FILTER_SANITIZE_STRING);
//this gets the the date and time in the sql format
$time = date("Y-m-d H:i:s");
```

Validation:

```
if ($babyID === null || $babyID === false ||
    $userID === null || $userID === false) {
    $error = "Invalid product data. Check all fields and try again.";
    echo $error;
}
```

Lessons Learned and possible future improvement:

I think it is pretty impractical that the person using this application would open their computer every time they want to create a log, so I think the best thing for a future improvement would be to make a version of this that could be a phone app. A phone is small and portable, so its easier to use to create quick logs so you don't forget them. Another improvement would be to add code so you can view the profiles that were created and have access to edit your own account as a user or doctor, or a user can edit a babys account. There could also be better reports that include things like growth percentiles if the user regularly records and update that information.