**Interview Summary**

*FILE HANDING AND ERROR HANDLING IN PHP*

\*Object

Real world entity, have attribute as variable and behavior as functions.

\*Class

Collection of objects, can be define as blueprint of object

\*Polymorphism

Performing one task into different ways, use method overloading and overriding to achieve it.

\*Abstraction

Hiding internal details and showing functionality. Use Abstract class and interface to achieve it.

Interface can have only abstract methods.

An abstract class can have abstract and non-abstract methods.

\*Encapsulation

Binding code and data together into single unit.

**\*Inheritance**

One class is allowed to inherit the features (fields and methods) of another class. Provides code reusability.

BASE CLASS ->extends DERIVED CLASS

Multiple not supported by php. (use interface or traits)

a + b

||

c

Multi level is supported.

a

|

b

|

C

**PROJECT**

**Mini Project - Blog It --- CSS, HTML, Bootstrap, JSON Api**

- It is a blogging website where user can post and read blogs of other users.

-User can add photos in the blog

-User can also search blogs according to different catagory or by words.

-For writing the blog user have given the text editor with features of different fonts and different sizes.

-We also add the functnality of comment.

-User can read the some part of the blogs without getting signup, but for reading the whole blog user have to login.

-For login and signup i use modals.

**Major Project - ShoutBox --- Apache server, Mysql Database, Laravel Framework, Angular 8**

-We have created a site where user can post images, video, and text.

-User can make new friends by accepting and sending the request.

-User can add or update there details in profile.

-User can see the post of friends and react on them by liking or commenting on them.

-User can also report the post of other user.

**ADMIN**

-There was 2 module Admin and User.

-Admin have to accept the request of the new user then only user can login into the website.

-Admin can delete the repoted Post and Users.

-Admin can see all the Users and their posts.

**Database**

-We created Single table for user having column for role Admin and User, in controller logic using if condition i was checking if the user is admin or not.

**VALIDATION**

-required, minimum,maximum,email,password

**What I did ?**

-My part was doing front end and back end of Registration and login

-I made 2 componets for Login and signup in angular and user controller and user Model in backend laravel.

-In backend we were using Laravel Eloquent ORM (Object Realation Mapper) Ex: if we make table name users then the model name will be user.

-In user controller i wrote query using Query builder for fetching data and send that data in Json api.

**Api**

-We were using api to connect front end with backend, so i made user services where i was providing the url of api and from there data will be fetched from database

**Approach new Technology**

Studing from basiscs

study from udemy courses

practice and doing hands-on

**Optimize the code**

Sonar scanner and test cases.

**Laravel Eloquent**

Eloquent ORM (Object Relation Mapper) provides an easy way to communicate with Database.

Can work with multiple databases efficiently.

It is an architectural pattern where the model created in the MVC structure corresponds to a table in the database.

Ex: if we make table name users then the model name will be user.

Using Eloquent ORM Model can perform common queries and can insert data into the database using seeder.

All we have to do is to define the database tables and relation between them, and Eloquent will do the rest.

We can perform CRUD operations with the Eloquent ORM

1. Create - ::create()

2. Retriving - ::all() ::find() ::where()->first() ::where()->get()

3. Updating - find -> update -> save()

4. Deleting - find->delete() find->destroy()

**MVC** -

Model : Stores and Manages data

View : User Interface (Visual Representation of data, user direct interact with view)

Controller : Business Logic, it connect the model and view (The controller converts inputs from the view to retrieve/update data in the model)

Flow

view -> controller -> Model

|

view <- Controller <-

**OVER LOADING & OVERRIDING**

Overloading : When two or more methods in the same class have the same name but different parameters, it's called Overloading or Compile Time polymor.

Overriding : When the method signature (name and parameters) are the same in the superclass and the child class, it's called Overriding or Run Time Polymor.

**Use of .htaccess and php.ini files in PHP?**

Both of them are used for making changes to the PHP settings.

.htaccess – A special file that can be used to change or manage the behavior of a website. Directing all users to one page and redirecting the domain’s page to https or www are two of the most important uses of the file. For .htaccess to work, PHP needs to be installed as an Apache module.

php.ini – This special file allows making changes to the default PHP settings. Either the default php.ini file can be edited, or a new file can be created with relevant additions and then saved as the php.ini file. For php.ini to work, PHP needs to run as CGI.

**Explain the difference between GET and POST requests.**

**Answer:**Any PHP developer needs to have an adequate understanding of the HTTP protocol. The differences between GET and POST are an indispensable part of the HTTP protocol learning. Here are the major differences between the two requests:

* GET allows displaying the submitted data as part of the URL. This is not the case when using POST as during this time, the data is encoded in the request.
* Typically, GET is used for retrieving data while POST is used for inserting and updating data.

#### What are the Traits in PHP?

**Answer:**The mechanism allows for the creation of reusable code in PHP-like languages where there is no support for multiple inheritances. A trait can’t be instantiated on its own.

## **PHP Implode Function**

## The implode function in PHP is used to "join elements of an array with a string".

* The implode() function returns a string from elements of an array. It takes an array of strings and joins them together into one string using a delimiter (string to be used between the pieces) of your choice.
* The implode function in PHP is easily remembered as **"array to string"**, which simply means that it takes an array and returns a string. It rejoins any array elements and returns the resulting string, which may be put in a variable.

## **PHP Explode Function**

* The explode function is used to**"**Split a string by a specified string into pieces i.e. it breaks a string into an array".
* The explode function in PHP allows us to break a string into smaller text with each break occurring at the same symbol. This symbol is known as the delimiter.
* Using the explodecommand we will create an array from a string. The explode() function breaks a string into an array, but the implode function returns a string from the elements of an array.

## **What are Magic Methods in PHP?**

Methods that begin with 2 underscores(\_\_) are generally called Magic methods in PHP. These methods names are limited to some list of php supported keywords that are reserved. So any function should not be defined with the name of PHP magic methods.

Usually, these functions should be defined by the user and there is no need to call them explicitly.

## **List of Magic Methods in PHP**

* \_\_construct()
* \_\_destruct()
* \_\_call($fun, $arg)
* \_\_get($property)
* \_\_set($property, $value)
* \_\_isset($content)
* \_\_unset($content)

**Joins**

* (INNER) JOIN: Returns records that have matching values in both tables
* LEFT (OUTER) JOIN: Returns all records from the left table, and the matched records from the right table
* RIGHT (OUTER) JOIN: Returns all records from the right table, and the matched records from the left table
* FULL (OUTER) JOIN: Returns all records when there is a match in either left or right table

