Ready Pizza

# Key Challenges

Below are the key challenges that I faced during creation of Ready Pizza portal and how I overcame the same

## Maintaining User types

There will be two kinds of users in the system namely, admin and customer. So, it was confusing to maintain the same.

So, I went ahead with creating one single table called tbl\_users and keeping a column in it called type where values can be user or admin. At the time of user registration, I am adding user in the type

Column. And at the time of user login I am checking the type column for value user. Also, I created all admin related php files in /admin folder and user related files in the base folder to separate the functionality.

## Maintaining Menu and header

As I was progressing and adding new pages, I had to change the menu to reflect the same. I had to go on each of the page and change the menu.

So, I decided to make a header php file that will contain the menu and imported the php file in all the pages. This way whenever I have to change the menu or any header item then I have to just change the header php file and it gets reflected in all the pages.

## Saving Pizza Images

I had created option in admin panel where admin can add and edit pizzas. They can fill the pizza details like pizza name, price and image. The image was not getting uploaded.

There was a problem with the folder privilege and the php file was not having access to write the uploaded image file to the images folder. So, I had researched on the same and provided write access to the folder and then it was successfully able to save the images.

## Checking of orders

I had created one page to list all orders in the admin panel. But after creating few orders I saw that it was hard to track current orders as there were many orders.

So, I went ahead and created two pages, one as current order and one as past orders. The current order page will show only the orders in ordered and prepared state and the past orders will show all the orders in delivered state. Generally, the admin will be concerned with checking and fulfilling the current orders, and this will help him not getting overwhelmed with data.

## Ordering multiple pizzas and toppings

Since the user can order multiple pizzas and toppings in a single order I was not sure how to save the total order in one table as the table can contain only predefined number of columns. I thought to create many columns like orderitem1, orderitem2…and so on. But this would have limited the capability of the system.

So, I went ahead and divided the orders in two tables, namely tbl\_orders and tbl\_orderitems. The tbl\_orders will contain only the order id, user id of the user who ordered and the timings. All the ordered items will be stored in the tbl\_orderitems in one row for each of the items. This way any number of items can be saved in as many rows in the database.

## Multiple Database Calls

As I was creating the pages I saw that many functions were called repeatedly like get user details etc. And I had to write all the functions again in each of the page. Also had to put database details in each of the page and whenever I had to change password etc. I had to do in all the pages and it was cumbersome and time consuming.

So, I went ahead and created a separate php file with all the database related interactions namely DataAccessLayer.php. It has the database connection details and all the other php pages call the functions inside this file to access the database. This way the functions got reused reducing the rework. And also, whenever database password etc. changes was to be done, it could be done in the same single file.