

Online food ordering site

by

Sara Ahmadian

Thesis submitted for the bachelor of computer
engineering (software)

Supervisor(s): Professor Dr. Adel Ghazikhani

Imam Reza International University

Summer 2021

Abstract:

While electronic commerce is growing rapidly, the food industry has also found its place in this growing area. Today, with the emergence of new business models, the traditional definitions of marketing have changed. Marketing is one of the organizational systems or one of the most important processes of commercial organizations in order to achieve the goals and satisfy the needs of consumers, therefore, the developments of the communication era in the form of the effects of the development of electronic commerce have caused significant changes in marketing. The study focuses on investigating the attitudes of restaurant consumers towards online food purchasing. Confirmed hypotheses affirm the significant and positive impact of factors such as the ease of the ordering process, perceived usefulness, innovations, promotions, and trust in online food ordering on consumer attitudes.

This project serves as an Online Food Ordering System, showcasing the ability to analyze requirements, design a user-friendly interface, and implement a functional web application. It effectively addresses challenges associated with traditional food ordering, providing a contemporary solution for both customers and restaurant owners. Key features include a user-friendly interface, efficient order management for restaurant owners, a secure payment gateway ensuring financial data integrity, and compatibility with major web browsers.

Keywords: consumer attitude, online shopping, consumer behavior, online food ordering system

Table of Contents:

1. Abstract -----	2
2. Chapter 1: Introduction and Getting to Know the Tools-----	5
3. Chapter 2: Analysis and Design of Diagrams-----	9
3.1. ER Diagram -----	9
3.2. Use Case Diagram-----	10
3.3. Activity Chart-----	11
3.3.1. Manager -----	11
3.3.2. User -----	12
3.4. Class Diagram -----	13
4. . Chapter 3: Implementation of the Codes -----	14
4.1. The Site Template is Divided into 4 General Sections -----	14
4.2. register.php Page -----	14
4.3. login.php Page -----	22
4.4. index.php Page -----	26
4.5. navbar.php Page -----	29
4.6. page about-us.php-----	28
4.7. page footer.php -----	29
4.8. categories.php Page -----	30

4.9. logout.php Page -----	31
4.10. foods.php Page -----	32
4.11. food-list.php Page -----	33
4.12. food-delete.php Page -----	35
4.13. food-add.php Page -----	36
4.14. Manager Table -----	37
4.15. Food Categories Table -----	37
4.16. Foods Table -----	38
4.17. Comments Table -----	38
4.18. Orders Table - -----	39
4.19. Users Table -----	39
5. Chapter 4: Examination of Various Site Pages -----	40
5.1. User Panel -----	40
5.2. Admin Panel -----	48
6. References -----	53

Chapter 1: Introduction and Familiarity with Tools

MySQL is a database management system that allows us to create and manage databases. A database can contain one or more tables, each of which can have various columns and rows to organize information in a structured manner.

Almost all software finds its real power through a database, and without it, they cannot perform specific tasks. For example, a sports club management system requires a database of individuals who have registered with it.

This database must have several different tables, all of which are placed in your database named "MYDATABASE." Each table also includes multiple columns or fields.

PHP programmers usually have two ways to access the MySQL database:

Using PHPMyAdmin

Using PHP coding

PHPMyAdmin is a tool used for manual management of MySQL, allowing operations such as creating a new database, constructing tables, defining columns, modifying table properties, and more. Almost anyone with any level of programming knowledge can interact with its interface and address their needs.

PHPMyAdmin is often pre-installed in server virtualization programs like XAMPP. To perform these tasks, open the XAMPP control panel, start APACHE and MYSQL, and proceed with the desired operations.

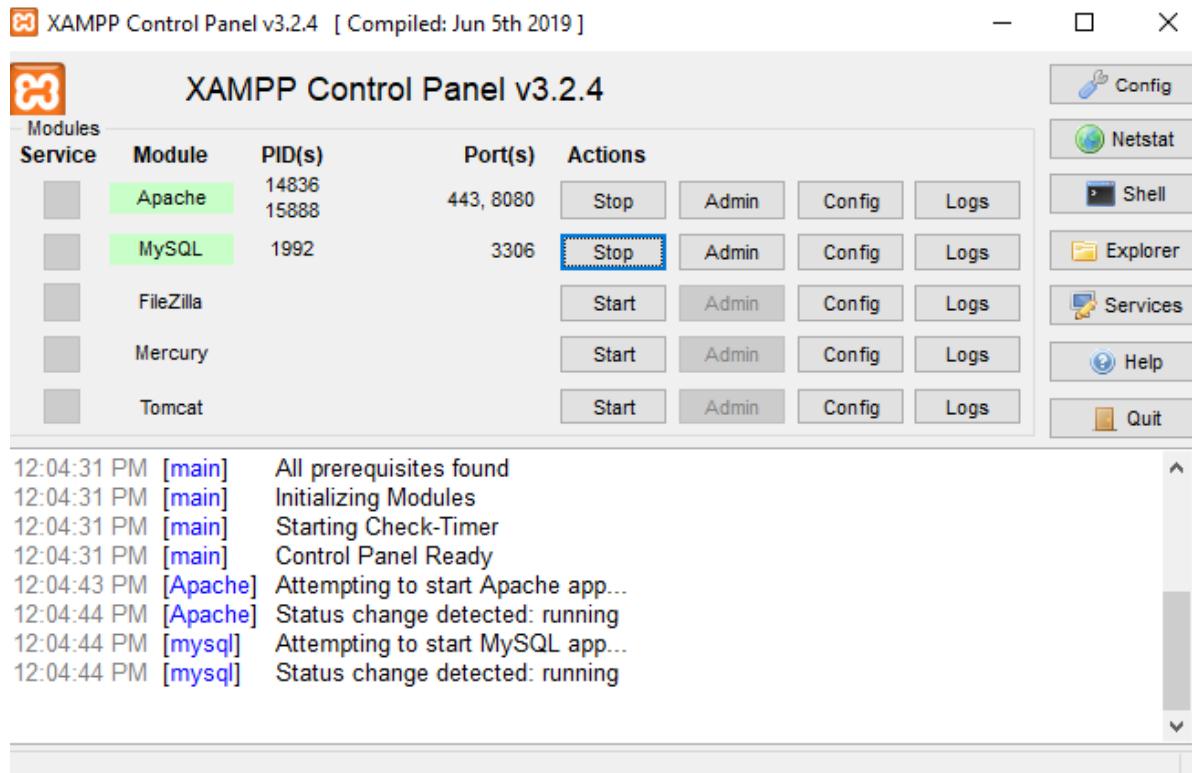


Figure 1-1: XAMPP Software Interface

After running the server program and activating the servers (similar to the image above), open your web browser. Type the expression "localhost/phpmyadmin" into the address bar of your browser and press Enter.

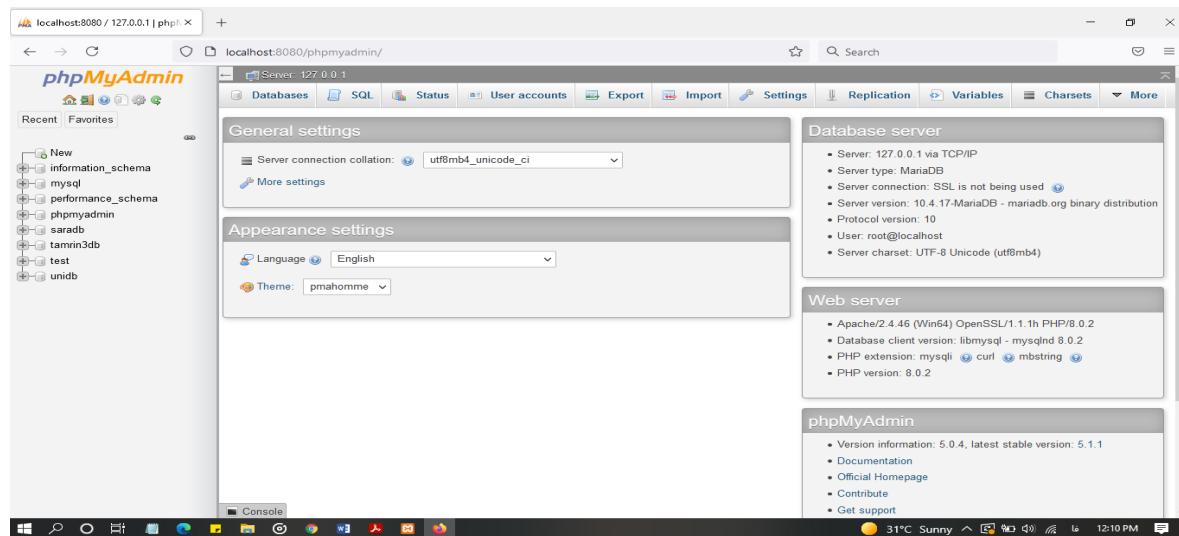


Figure 2-1: Accessing the PHPMyAdmin Page

Then, to create a new database, click on the "NEW" option.

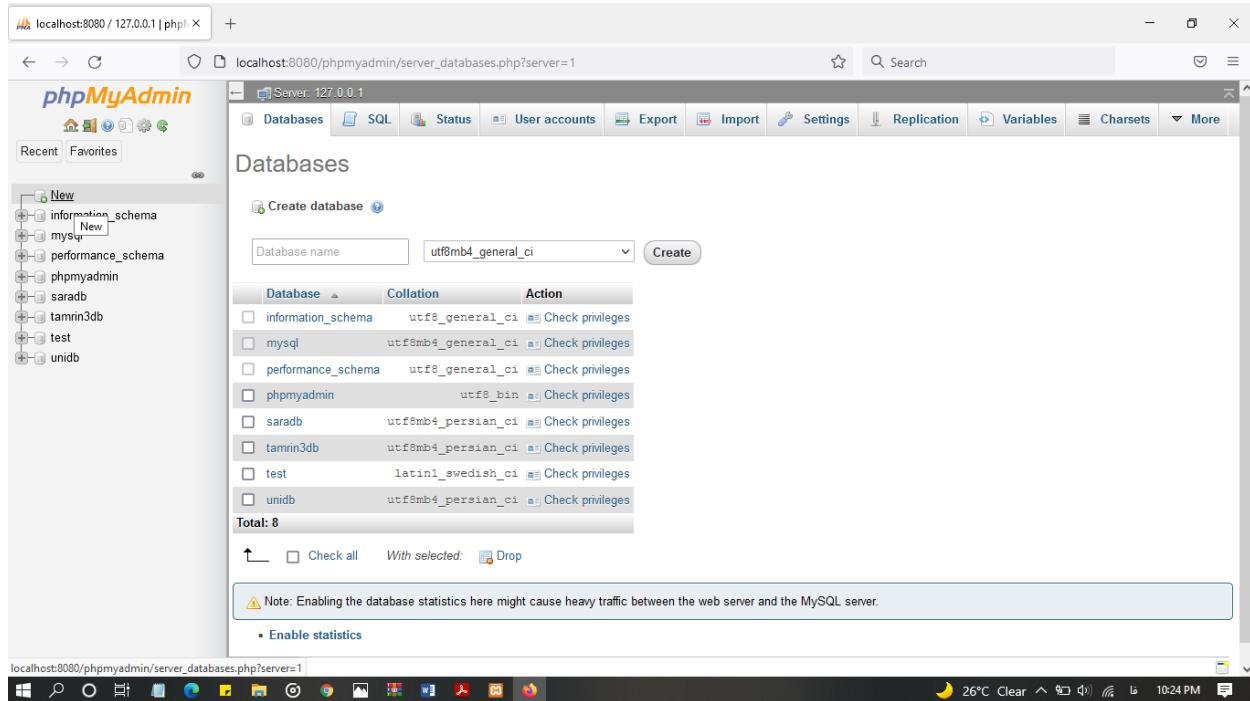


Figure 3-1: Creating a New Database

In the "Database name" section, enter the name for your database. From the "Collation" section, choose the Unicode (UTF-8) encoding for your database. Finally, click on the "Create" button to complete the process.

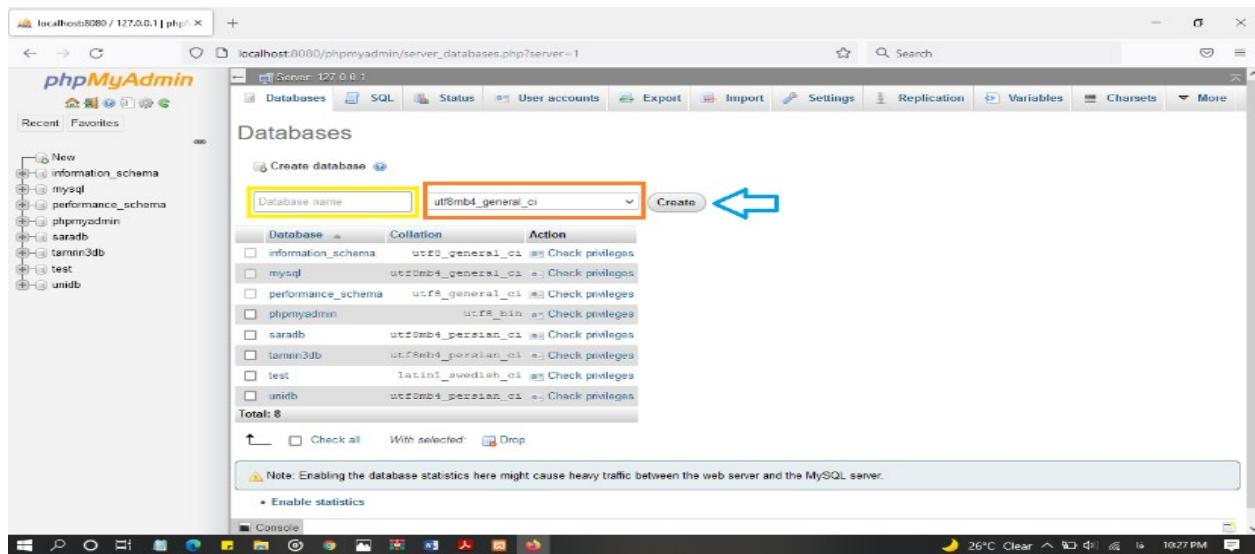


Figure 4-1: Steps for Registering a New Database

MySQL, as a database management system, possesses features such as ease of management, high performance, scalability, compatibility, complete data security, cost-effectiveness, and excellent integration with the PHP programming language.

Queries, in essence, are commands given to the database, instructing it to perform one of the following tasks:

- Delete a specific piece of data.
- Edit a portion of data.
- Add new data.
- Search for new data.

SQL (Structured Query Language) is a programming language that can:

- Execute queries on the database.
- Read data from the database.
- Insert data into the database.
- Edit and update existing data in the database.
- Delete existing data from the database.
- Create a new database.

In this project, Adobe Dreamweaver software has been utilized for coding purposes.

Chapter 2: Analysis and Design of Diagrams

Entity-Relationship (ER) Diagram:

The Entity-Relationship (ER) diagram is comprised of a series of graphical symbols used to represent the data structure of the system under consideration. This diagram is employed to visually depict the behavioral aspects of the database.

The ER diagram illustrates the behavior of the database graphically, providing a visual representation of the data structure of the system.

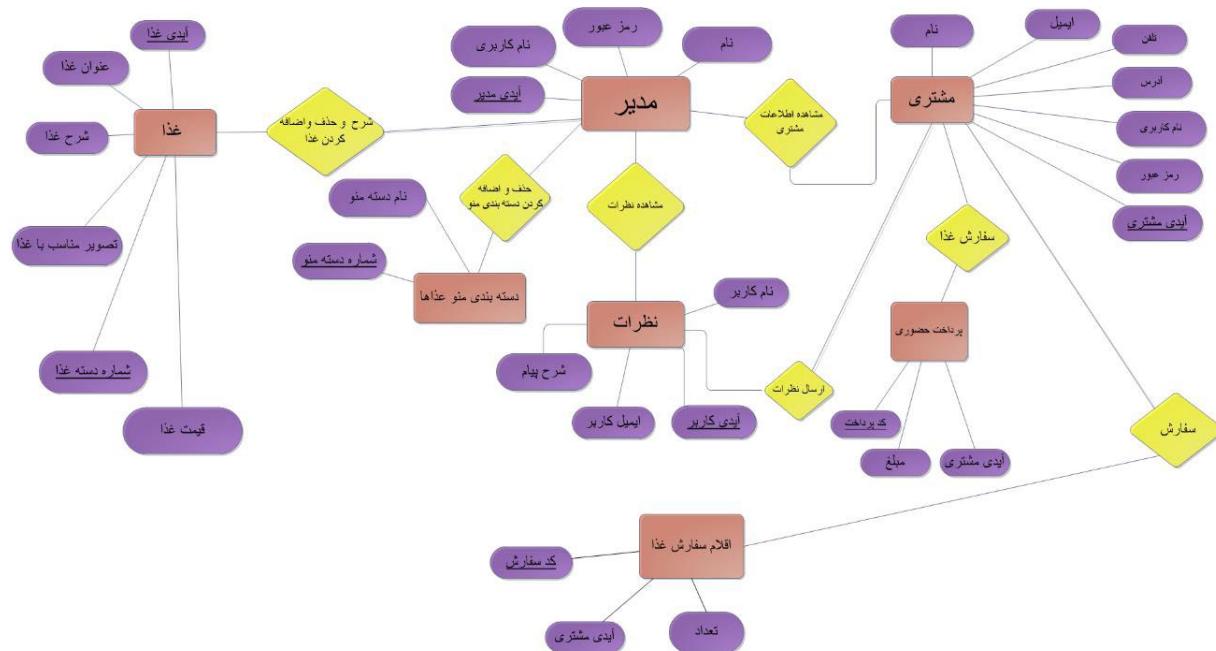


Figure 1-2: Entity-Relationship (ER) Diagram

Use Case Diagram:

The Use Case Diagram for this website consists of two main sections: Customer and Manager. It depicts the actions that each user type can perform, showcasing key operations, functionalities, or features of the system under consideration. The diagram visually represents the critical interactions between users and the system, outlining the scope and functionality of the website for both customers and managers.

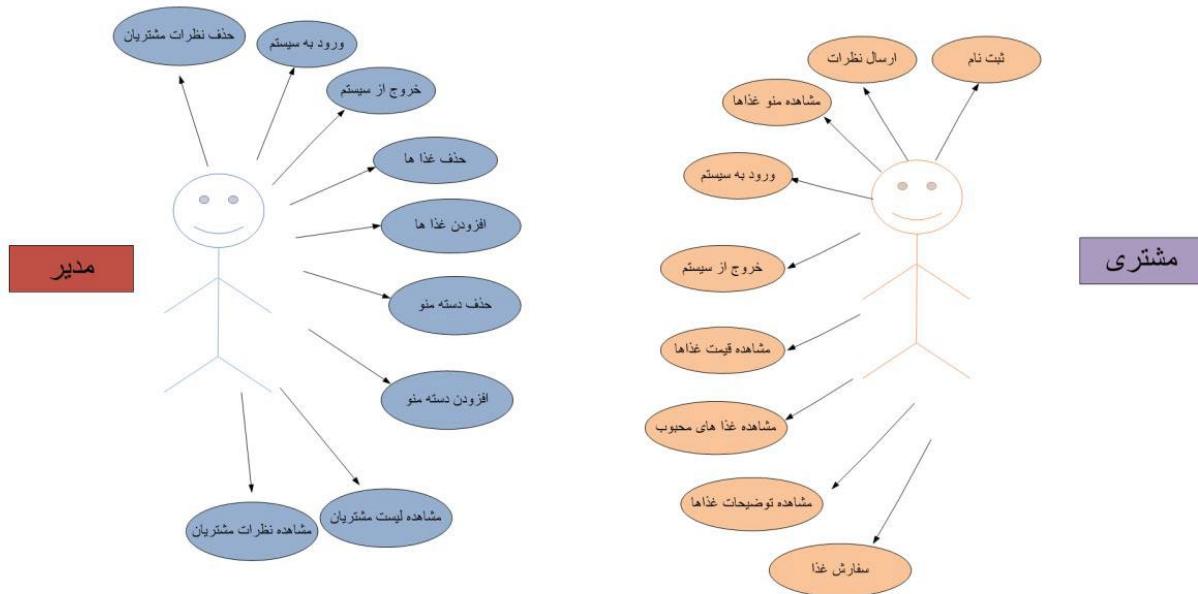


Figure 2-2: Use Case Diagram

Activity Diagram:

Manager

The activity diagram illustrates the graphical representation of the steps a manager takes from logging into the site to logging out. In other words, it provides a visual depiction of the processes, tasks, and logical execution flow of the manager within the system.

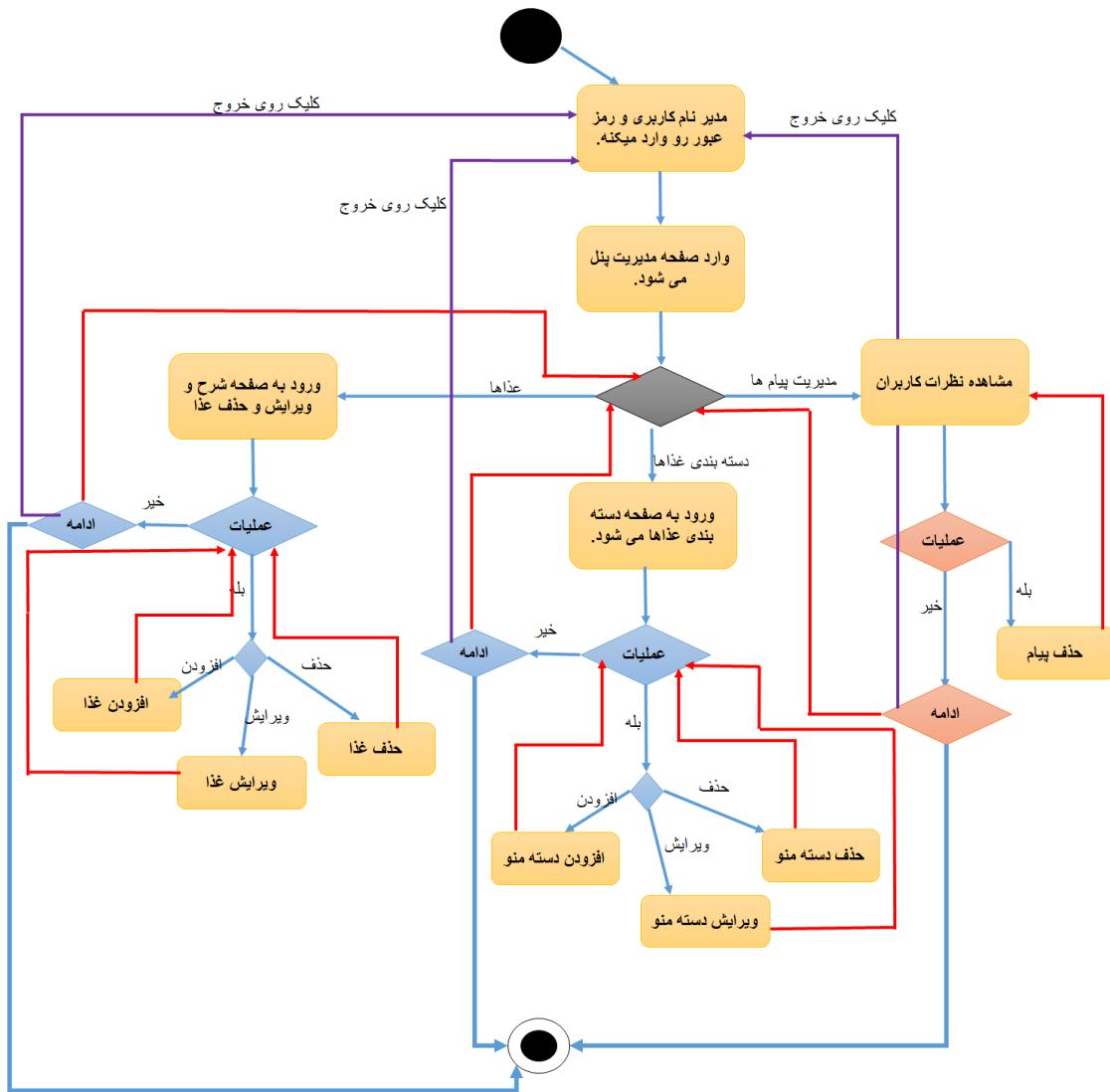


Figure 3-2: Manager Activity Diagram

User

The activity diagram illustrates the graphical representation of the steps a user takes from logging into the site to logging out. In other words, it provides a visual depiction of the processes, tasks, and logical execution flow of the user within the system.

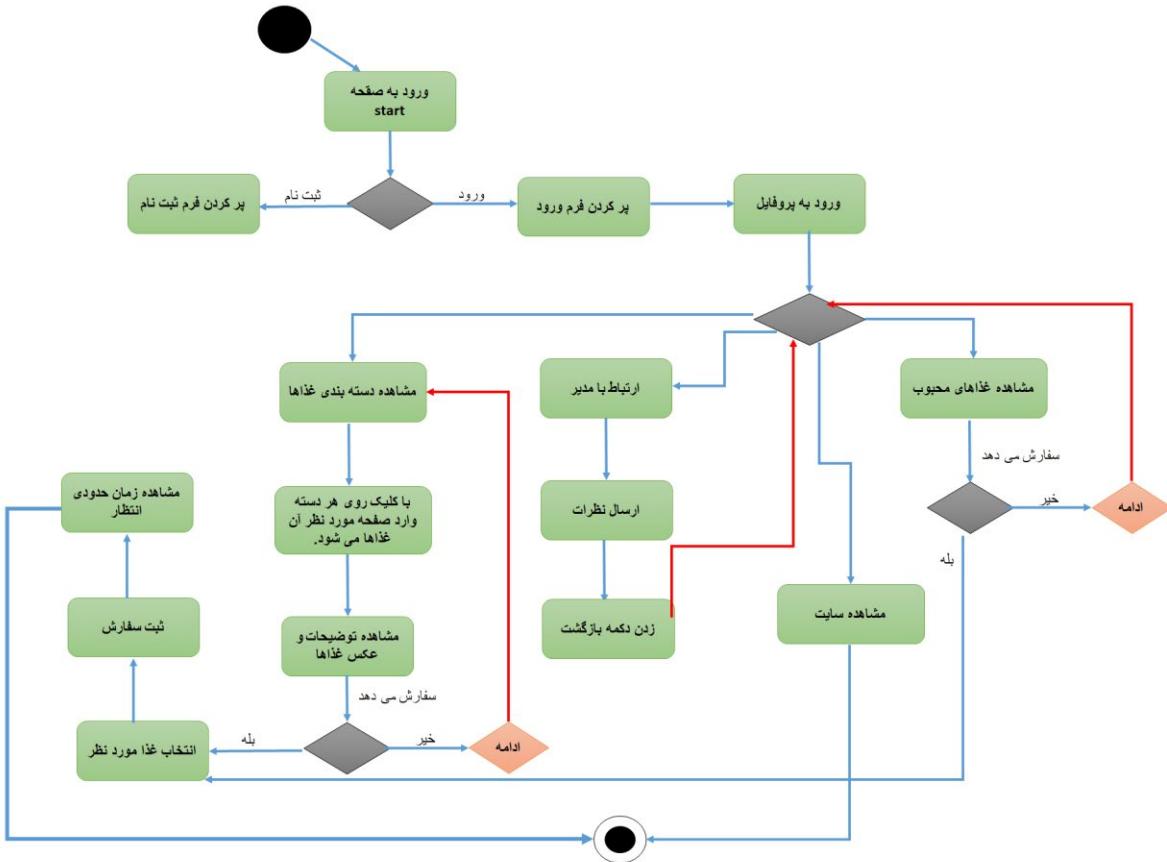


Figure 4-2: User Activity Diagram

Class Diagram:

The Class Diagram plays a significant role in generating high-quality code. It is considered the primary source for code generation. The diagram indicates that a manager can establish relationships with multiple users, while each user can only communicate with one manager. This visualization helps in understanding the structure of the code and the relationships between different classes in the system.

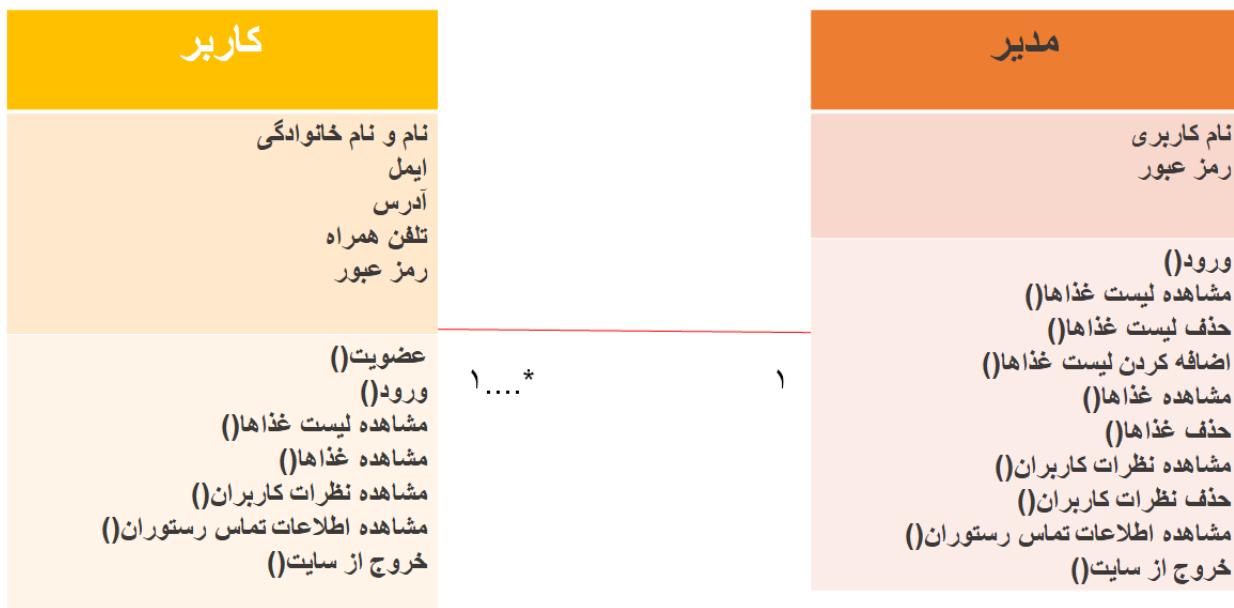


Figure 5-2: Class Diagram

Chapter 3: Implementation of Codes

The website template is divided into four main sections:

1. Header: This section is static and remains constant on some pages of the site.
2. Site Content: This dynamic section changes for each page.
3. Sidebar: This section is static on some pages of the site.
4. Footer: This section is static on some pages of the site.

As an example, let's consider the implementation of the code for the "register.php" page.

1)register.php

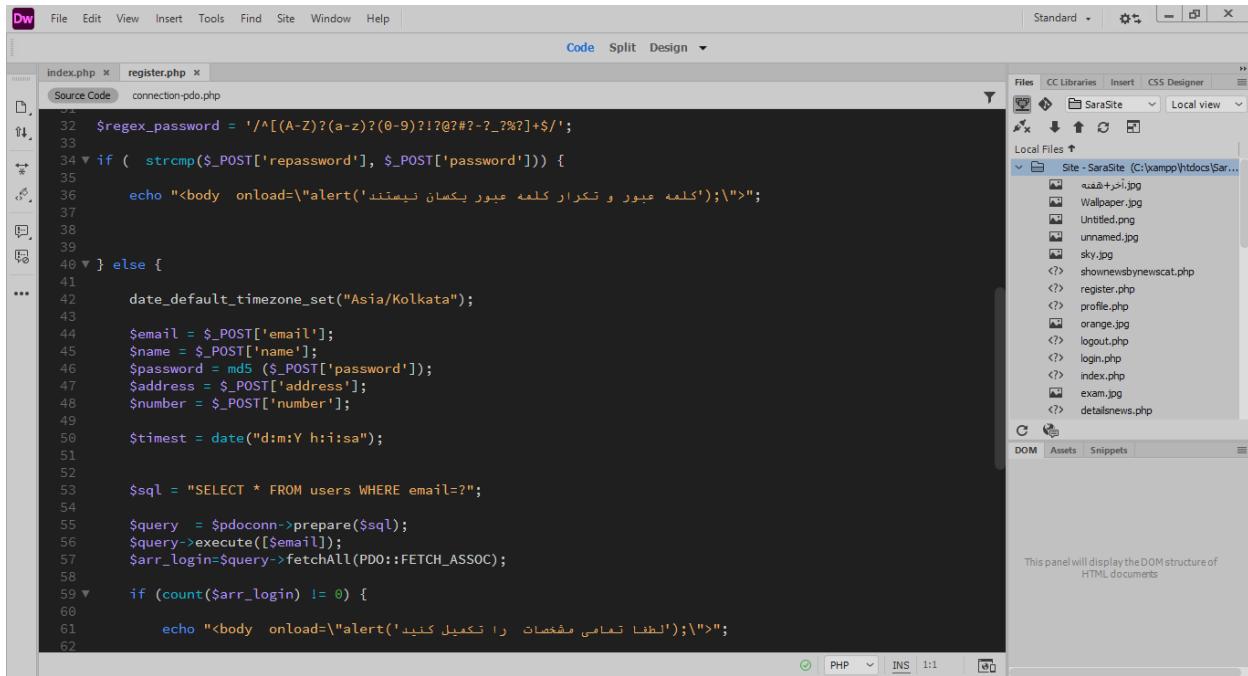
The screenshot shows the Adobe Dreamweaver interface. The top menu bar includes File, Edit, View, Insert, Tools, Find, Site, Window, Help, Standard, and Local view. The toolbar below has icons for Undo, Redo, Cut, Copy, Paste, Find, Replace, Select All, and others. The code editor window displays the following PHP code for register.php:

```
1 <?php
2
3
4 try {
5
6     if (!file_exists('connection-pdo.php'))
7         throw new Exception();
8     else
9         require_once('connection-pdo.php');
10
11 } catch (Exception $e) {
12
13     echo "<body onload=\"alert('"; // مشکلی در سرور وجود داشت! لطفاً بعد از حدتی دوباره امتحان کنید'";
14
15 }
16
17 if (!isset($_POST['name']) || !isset($_POST['email']) || !isset($_POST['address']))
18     || !isset($_POST['number'])|| !isset($_POST['password']) || !isset($_POST['repassword'])) {
19
20     echo "<body onload=\"alert('"; // انتخیر ها نامعتبر لطفاً صفحه را مجدداً بارگذاری کنید'";
21
22 }
23
24
25
26
27
28
29 $regex_email = '/^[_a-zA-Z0-9-]+(\.[_a-zA-Z0-9-]+)*@[a-zA-Z0-9-]+\.([_a-zA-Z0-9-]+)*(\.[a-zA-Z]{2,3})$/';
30 // $regex_name = '/^[(A-Z)?(a-z)?(0-9)?\s*]+\$/';
```

The right side of the interface shows the "Files" panel with a tree view of the website structure under "Site - SaraSite". It lists several files including index.php, connection-pdo.php, register.php, profile.php, orange.jpg, logout.php, login.php, index.php, exam.jpg, and detailsnews.php. Below the files is a "DOM" panel which displays the DOM structure of HTML documents.

Figure 1-3: Portion of the register.php Page

A portion of the code that establishes the server connection to the specified database, and if the connection is not established, an error message is displayed.



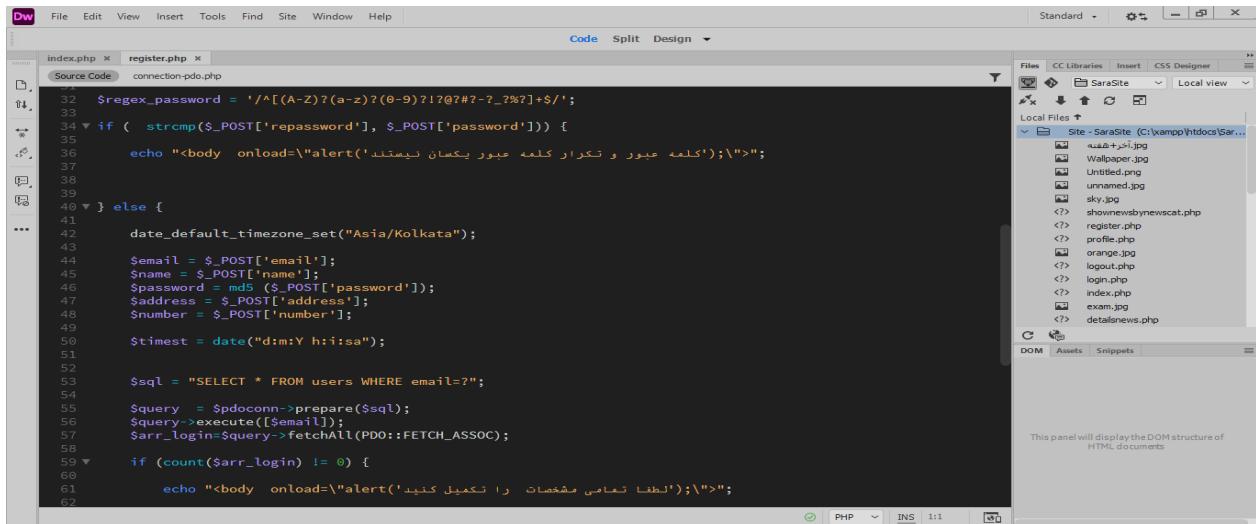
```

32 $regex_password = '/^[(A-Z)(a-z)(0-9)!@#?-?%?]+$/';
33
34 if (strcmp($_POST['repassword'], $_POST['password'])) {
35     echo "<body onload=\"alert('كلمة عبور و تكرار كلمة عبور يكملان بحسبت');\">";
36 }
37
38
39
40 } else {
41
42     date_default_timezone_set("Asia/Kolkata");
43
44     $email = $_POST['email'];
45     $name = $_POST['name'];
46     $password = md5($_POST['password']);
47     $address = $_POST['address'];
48     $number = $_POST['number'];
49
50     $timest = date("d:m:Y h:i:sa");
51
52
53     $sql = "SELECT * FROM users WHERE email=?";
54
55     $query = $pdoconn->prepare($sql);
56     $query->execute([$email]);
57     $arr_login=$query->fetchAll(PDO::FETCH_ASSOC);
58
59 if (count($arr_login) != 0) {
60
61     echo "<body onload=\"alert('إخطار تعامى معلومات را تكميل كنيد');\">";
62

```

Figure 2-1-3: Continued Code of the Register Page

This section pertains to validating the user's password during the registration process, enhancing security.



```

32 $regex_password = '/^[(A-Z)(a-z)(0-9)!@#?-?%?]+$/';
33
34 if (strcmp($_POST['repassword'], $_POST['password'])) {
35     echo "<body onload=\"alert('كلمة عبور و تكرار كلمة عبور يكملان بحسبت');\">";
36 }
37
38
39
40 } else {
41
42     date_default_timezone_set("Asia/Kolkata");
43
44     $email = $_POST['email'];
45     $name = $_POST['name'];
46     $password = md5($_POST['password']);
47     $address = $_POST['address'];
48     $number = $_POST['number'];
49
50     $timest = date("d:m:Y h:i:sa");
51
52
53     $sql = "SELECT * FROM users WHERE email=?";
54
55     $query = $pdoconn->prepare($sql);
56     $query->execute([$email]);
57     $arr_login=$query->fetchAll(PDO::FETCH_ASSOC);
58
59 if (count($arr_login) != 0) {
60
61     echo "<body onload=\"alert('إخطار تعامى معلومات را تكميل كنيد');\">";
62

```

Figure 3-1-3: Continued Code of the Register Page

Functions on the site are called by different names. All the necessary functions for the site are defined and stored in this file so that they can be used wherever needed.

The screenshot shows the Adobe Dreamweaver interface with the following details:

- File Menu:** File, Edit, View, Insert, Tools, Find, Site, Window, Help.
- Toolbar:** Standard, Code, Split, Design.
- Code Editor:** The "register.php" file is open in Source Code view. The code handles user login and registration. It uses PDO for database connection and preparation of SQL statements. It includes validation logic for email and password fields, and inserts new users into the "users" table.
- File Explorer:** Shows the local files structure under "Site - SaraSite".
- DOM Panel:** Displays the DOM structure of the HTML document.

```
index.php x register.php x
Source Code connection-pdo.php

49
50     $timest = date("d:m:Y h:i:s");
51
52
53     $sql = "SELECT * FROM users WHERE email=?";
54
55     $query = $pdoconn->prepare($sql);
56     $query->execute([$email]);
57     $arr_login=$query->fetchall(PDO::FETCH_ASSOC);
58
59     if (count($arr_login) != 0) {
60
61         echo "<body onload=\"alert('الطباطا تسامي مشخصات را تکمیل کنید');\">";
62
63
64
65
66 } else {
67
68     $sql = "INSERT INTO users(`name`, `email`, `password`, `timestamp`, `address`, `number`)"
69     VALUES(?, ?, ?, ?, ?, ?)";
70     $query = $pdoconn->prepare($sql);
71     if ($query->execute([$name, $email, $password, $timest, $address, $number]))
72     {
73
74         echo "<body onload=\"alert('مشخصات کاربری شما با موقتیت ثبت شد. از قسمت ورود اقدام کنید');\">";
75
76
77
    
```

Figure 4-1-3: Continued Code of the Register Page

The code in this section forces the user to fill in all the fields in the registration form; otherwise, they will encounter an error message, and the registration will not proceed.

Figure 5-1-3: Continued Code of the Register Page

If the information is complete, it saves them to the database through the mentioned code. Regular expressions define patterns using a sequence of symbols, letters, or special characters, each with a specific meaning.

Character "/": The regular expression is placed between two slashes.

Character "^": The regular expression starts with the hat (^) character, followed by a letter indicating the string's start.

Character "\$": The regular expression ends with the dollar (\$) character, preceded by a letter indicating the string's end.

Characters "][": A range of characters inside brackets is defined.

Characters "{}": The number of characters in the preceding expression inside the curly braces specifies the quantity.

Character """: The asterisk () character means zero or more repetitions of the preceding character.

Character "+": The plus (+) character means one or more repetitions of the preceding character; it requires at least one repetition.

Character "\w": It represents all Latin alphabet letters and numbers 0 to 9.

Character "\s": It signifies any whitespace character, such as space or newline.

href: Indicates the location and name of our CSS file.

rel: Specifies the relationship type between HTML and CSS.

type: Represents the type of added content.

alert: Used on websites to display an alert message to the user, indicating that they have entered an incorrect value or that the operation was not successful.

regex: Stands for regular expressions, involving the use of pattern-matching expressions.

In the PHP language, the following functions are used for working with regular expressions:

preg_match

preg_match_all

preg_replace

preg_split

preg_quote

"[A-Z[": Includes a string containing at least uppercase English alphabet letters A to Z. For example, it does not include abc, 123, aa, etc.

"[a-z[": Includes a string containing at least lowercase English alphabet letters a to z. For example, it does not include 1@2, JAVASCRIPT, 123, etc.

"[0-9[": Includes a string containing at least the digits 0 to 9. For example, it does not include abcd, p%@&*, aa, etc.

"[a-zA-Z^["": Includes a string that, except for lowercase letters a to z, may contain other characters. For example, it does not include php, javascript, bbbb, etc.

"[A-Z^["": Includes a string that, except for uppercase letters A to Z, may contain other characters. For example, it does not include PHP, JAVASCRIPT, BBBB, etc.

"[0-9^["": Includes a string that, except for digits 0 to 9, may contain other characters. For example, it does not include 000, 222, 123, etc.

\$sql: A variable of string type.

fetchAll: In fact, PDOStatement::fetchAll returns an array containing all the rows returned by our query.

Prepare: Using this command, you can create a template for an SQL query and send it to the database. The reason for using the term template here is that instead of necessary parameters in the query, we use a question mark (?).

Should we use MySQLi or PDO?

The choice between MySQLi and PDO depends on various factors, and both have their advantages:

Database Compatibility:

MySQLi: Specifically designed for MySQL databases.

PDO: Works with 12 different database systems, providing more flexibility. If you might need to switch to a different database system in the future, PDO makes this process easier.

API Style:

MySQLi: Object-oriented and procedural API.

PDO: Object-oriented only.

Ease of Use and Flexibility:

MySQLi: Requires a bit more code to accomplish some tasks compared to PDO.

PDO: Generally considered more flexible and easier to use, especially when dealing with multiple database systems.

Prepared Statements:

MySQLi: Supports both procedural and object-oriented approaches to prepared statements.

PDO: Provides a consistent and simpler interface for prepared statements across different database systems.

Security:

Both MySQLi and PDO support prepared statements and parameterized queries, offering protection against SQL injection when used correctly.

Community Support and Documentation:

MySQLi: Being MySQL-specific, it has extensive documentation and community support.

PDO: Also well-documented with good community support.

API Support:

MySQLi: Has more advanced features like asynchronous queries and embedded server support.

PDO: Generally has a more consistent interface but lacks some advanced features present in MySQLi.

In summary, if you are working exclusively with MySQL and prefer a more procedural approach, MySQLi might be a good fit. If you want a more flexible and consistent approach that supports multiple database systems, PDO is a strong choice. Consider your project requirements, coding style preferences, and potential future changes when making your decision.

2) login.php

The login page code is similar to the register page. A portion of the code establishes a connection between the server and the desired database, displaying an error if the connection is not established.

Figure 2-3: A section of the login page code

When we want to temporarily store values on the server side and use those values across different pages of a site, we use sessions. These values are accessible and viewable throughout all pages of the site.

Figure 1-2-3: Continuation of the login page code

One common use of sessions is during user login to a site. When you enter a site and browse its various pages, your username is accessible and visible on all pages.

To work with sessions, you first need to start them. To do this, you should call the session_start method. This should be written in the first line of the PHP code and before the HTML code.

When you want to store temporary information about your users so that it can be accessed on different pages of your website, you should use the `$_SESSION` variable. `$_SESSION` is the name of a global associative array used to temporarily store information about the current user until the session is active. For this purpose, you choose an arbitrary name as the key for the data you want to store.

Function count: In PHP, it recursively counts the elements of an array; meaning, it returns the number of elements present in the array.

Foreach loop: It works only on arrays, unlike other loops that work on any variable. Therefore, if used on other variables, an error occurs.

Dw File Edit View Insert Tools Find Site Window Help Standard Code Split Design

index.php x login.php x Source Code connection-pdo.php

Dynamically-related files cannot be discovered because there is no site definition for this document. Setup

```
44
45    if (count($arr_login) != 0) {
46
47        foreach ($arr_login as $key) {
48
49            $tmp_pass = $key['password'];
50            $tmp_name = $key['name'];
51            $tmp_id = $key['id'];
52
53        }
54
55        if ($tmp_pass == $password) {
56
57            session_start();
58
59            $_SESSION['user'] = explode(" ", $tmp_name)[0];
60            $_SESSION['user_id'] = $tmp_id;
61
62
63            echo "<body onload=\"alert('کاربر کرامی '$tmp_name شد');\">";
64
65
66        } else {
67            session_start();
68            echo "<body onload=\"alert('ایام کاربری یا کلمه عبور افتباشد');\">";
69
70        }
71
72    } else {
73
```

Files CC Libraries Insert CSS Designer Local Files Site - SaraSite (C:\xampp\htdocs\Sara... عکس پس زمینه.jpg Wallper.jpg Untitled.png unnamed.jpg sky.jpg <> shownewsbynewscat.php <> register.php <> profile.php orange.jpg <> logout.php <> login.php <> index.php exam.jpg <> detailsnews.php

DOM Assets Snippets

This panel will display the DOM structure of HTML documents

Figure 2-2-3: Continuation of the login page code

To use the foreach loop, there are two methods:

Method 1:

It takes a variable of type array and a variable like \$value. The \$value variable works in such a way that it assigns the value of each element of the array in turn. We can then use it in statements. This process continues until the array pointer is on the last element of the array, at which point the loop terminates.

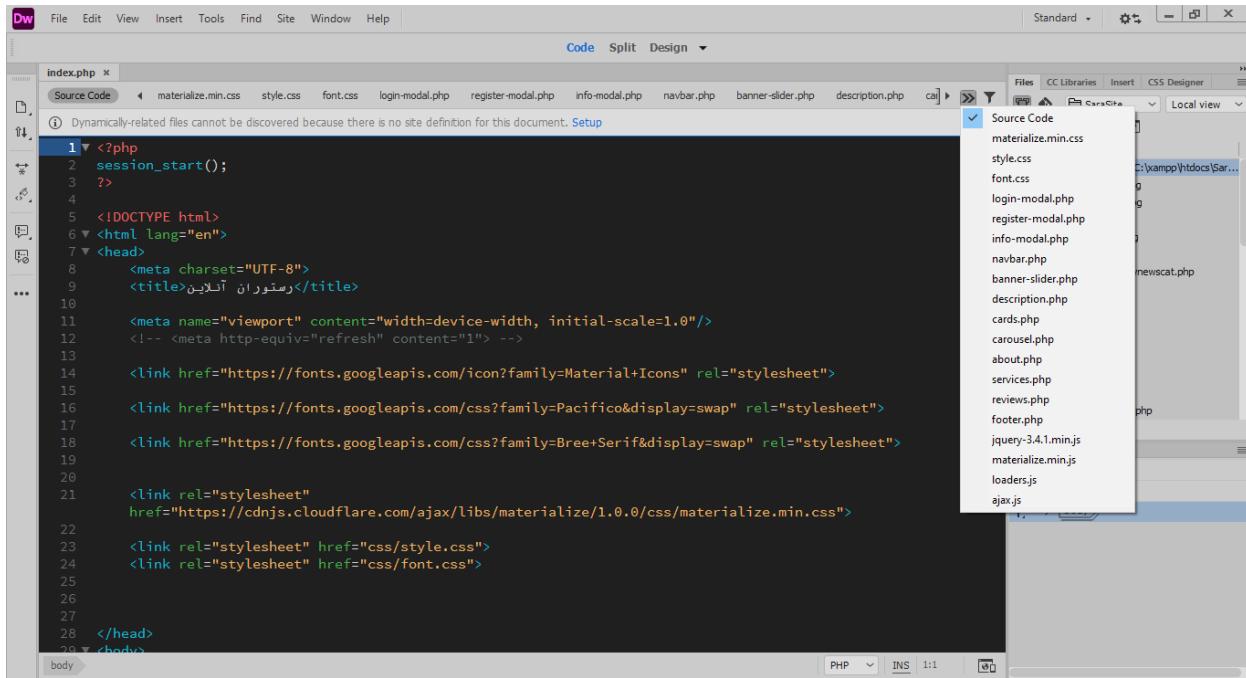
Method 2:

Similar to the previous method, it takes a variable of type array and two other variables, \$value and \$key. This means that, in addition to the value (\$value), we also have access to the key (\$key) of each element in the array.

Function explode: This function breaks down a textual string and places each part inside an array element. It takes three inputs. Firstly, the separator at which the string should be split. Secondly, the text that should be divided into different parts. Thirdly, an optional parameter that determines the number of elements to be returned by the function.

3)index.php

Index page is one of the most important and practical pages where the main layout of the website is placed.



The screenshot shows the Adobe Dreamweaver interface with the 'Source Code' tab selected. The code editor displays the following PHP and HTML code for the index page:

```
1 <?php
2 session_start();
3 ?
4
5 <!DOCTYPE html>
6 <html lang="en">
7 <head>
8     <meta charset="UTF-8">
9     <title>رسندر انجلین</title>
10    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
11    <!-- <meta http-equiv="refresh" content="1"> -->
12
13    <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
14
15    <link href="https://fonts.googleapis.com/css?family=Pacifico&display=swap" rel="stylesheet">
16
17    <link href="https://fonts.googleapis.com/css?family=Bree+Serif&display=swap" rel="stylesheet">
18
19
20    <link rel="stylesheet"
21        href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css">
22
23    <link rel="stylesheet" href="css/style.css">
24    <link rel="stylesheet" href="css/font.css">
25
26
27
28 </head>
29 <body>
```

The right panel shows the 'Files' panel with various files listed, including materialize.min.css, style.css, font.css, and other PHP files like login-modal.php and register-modal.php.

Figure 3-3, a portion of the code for the index page.

As you can see, various sections of the website are included on the index page. It contains PHP, CSS, and JavaScript codes.

3-1) navbar.php

The screenshot shows the Adobe Dreamweaver interface. The code editor on the left displays the PHP code for the navbar. The file browser on the right shows the local files for the website, including various PHP files like index.php, register.php, and login.php, along with image files like wallpaper.jpg and sky.jpg.

```
<?php
if (isset($_SESSION['user'])) {
    echo '<li><a href="#" class="hvr-grow" href="admin/index.php" data-target="modal1">پنل مدیریت</a></li>
    <li><a style="font-family: IRANSans;" href="#" data-target="modal2" href="onlineestoran/contact.php" class="hvr-grow modal-trigger" data-target="modal2">لایه اینلاین</a></li>
    <li><a style="font-family: IRANSans;" href="backends/message.php" class="hvr-grow" href="onlineestoran/about-restro-circle.php" data-target="modal3">نظرات</a></li>' ;
}
else {
    echo '<li><a href="#" class="hvr-grow modal-trigger" data-target="modal1" href="register.php" href="onlineestoran/contact.php" class="hvr-grow" data-target="modal1">ورود</a></li>
    <li><a href="#" class="hvr-grow modal-trigger" data-target="modal2" href="logout.php" href="onlineestoran/contact.php" class="hvr-grow" data-target="modal2">خروج</a></li>' ;
}
?>
```

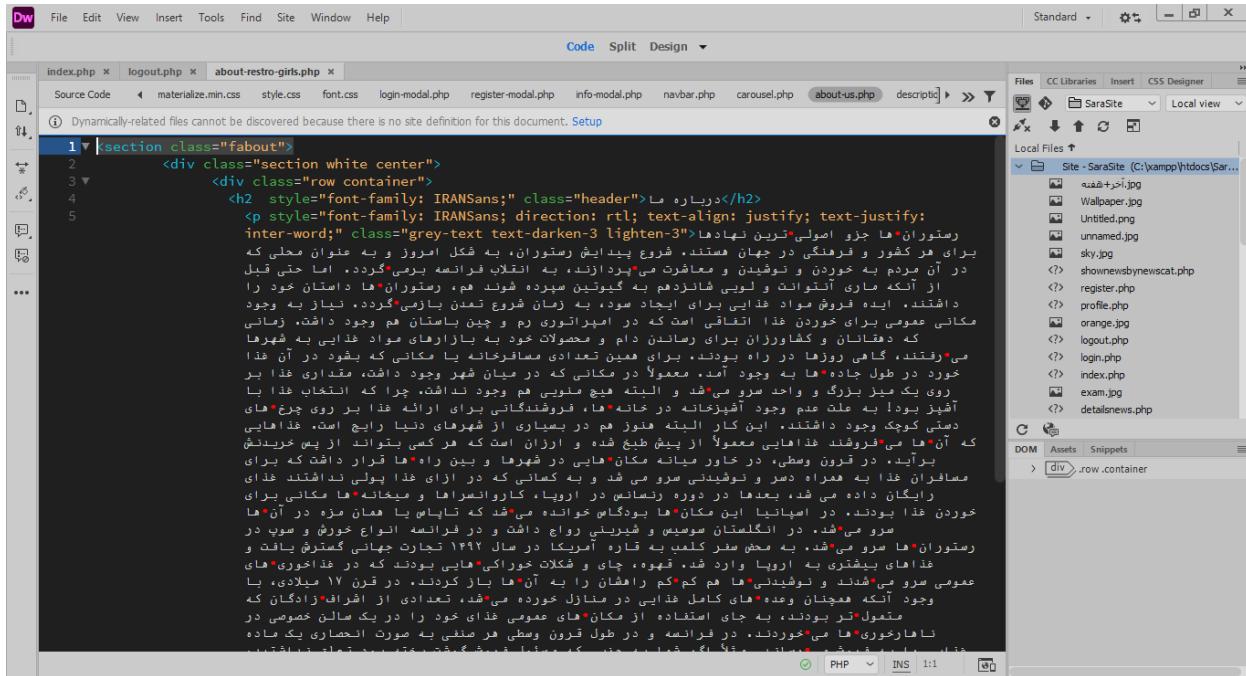
Figure 1-3-3, a portion of the code from the index page

One of its sections is related to the user profile page to which the user is redirected after logging in. The "navbar.php" page is the same page related to the user profile. After the user fills out and submits the login form and successfully logs in, they are directed to this page where they are welcomed.

Structurally and in terms of layout, it will be exactly the same as the index page, with the difference that, by receiving user information, they enter their personal page, and then the user can place their food orders.

3-2) about-us.php

Another page linked to the index page is the "About Us" page. This page provides information about your website, including its history, personnel, management, and anything else that may be useful for users to read.



The screenshot shows the Adobe Dreamweaver interface with the 'about-us.php' file open. The code editor displays the following content:

```
<?php
// PHP code for session handling
session_start();
if (!isset($_SESSION['user'])) {
    header("Location: login.php");
    exit();
}

// HTML and CSS code for the page structure
<html>
<head>
    <title>About Us - Restaurant Girls</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link href="style.css" rel="stylesheet" />
</head>
<body>
    <div class="header">
        <h1>Restaurant Girls</h1>
        <p>Your Local Restaurant</p>
    </div>
    <div class="content">
        <h2>About Us</h2>
        <p>Restaurant Girls is a local restaurant located in the heart of the city. We offer a variety of delicious dishes at reasonable prices. Our menu includes Italian, Mexican, and American cuisine. We are open from 11 AM to 9 PM, Monday through Saturday. Come visit us today!</p>
    </div>
</body>
</html>
```

The code editor also shows the file structure on the right, including files like index.php, logout.php, and other site files.

Figure 4-3, Code of the About Us Page

This page is comprised of a session and various HTML tags such as `<div>`, `<p>`, and `<h2>`. As you can see, there are descriptions about the website written on it.

3-3) footer.php

The screenshot shows the Adobe Dreamweaver interface. The code editor on the left displays the HTML code for the footer page. The code includes sections for restaurant details, social media links, and a copyright notice. The file browser on the right shows the local files structure for the website, including various PHP and image files.

```
1 <section class="ffooter">
2   <footer class="page-footer">
3     <div class="container">
4       <div class="row">
5         <div class="col l6 s12">
6           <h5 style="font-family: IRANSans; direction: rtl;" class="white-text">دریاره مکان</h5>
7           <p style="font-family: IRANSans; direction: rtl;" class="grey-text text-lighten-4">مکان آنلاین مشهد - رستوران آنلاین</p>
8           <p style="font-family: IRANSans; direction: rtl;" class="grey-text text-lighten-4">تلن 5130000000</p>
9           <p style="font-family: IRANSans; direction: rtl;" class="grey-text text-lighten-4">صریحه همراه روزه در طول این هفته</p>
10      </div>
11      <div class="col l4 offset-l2 s12">
12        <h5 style="font-family: IRANSans; direction: rtl;" class="white-text"> شبکه های اجتماعی</h5>
13        <ul style="font-family: IRANSans; direction: rtl;">
14          <li><a class="grey-text text-lighten-3" href="#">Facebook</a></li>
15          <li><a class="grey-text text-lighten-3" href="#">Instagram</a></li>
16          <li><a class="grey-text text-lighten-3" href="#">Twitter</a></li>
17          <li><a class="grey-text text-lighten-3" href="#">Whatsapp</a></li>
18        </ul>
19      </div>
20    </div>
21  </div>
22
23  </footer>
24 </section>
```

Figure 5-3, Code of the footer Page

The footer page, also known as the bottom or end section of the website, is dedicated to restaurant information. It includes details such as phone number, address, operating hours throughout the week, and relevant links for social media membership. This page is composed of a session and several HTML tags, including `<div>`, `<p>`, and `<h>`.

3-4) categories.php

The screenshot shows the Adobe Dreamweaver interface. The code editor window displays the PHP code for the 'categories.php' page. The code includes HTML structure and PHP logic for displaying food categories. The file browser on the right shows the local site structure under 'Site - SaraSite'. The DOM panel at the bottom shows the current selection path: section > .fcategories > div > .container > div > .section.white.center.

```
17 <section class="fcategories">
18
19   <div class="container">
20
21     <div class="section white center">
22       <h3 style="font-family: IRANSans" class="header">دسته بندی غذا</h3>
23     </div>
24
25 <?php if (count($arr_all) == 0) {
26   echo '<div class="section gray center" style="border: 1px solid black; border-radius: 5px;">
27     <p style="font-family:IRANSans" class="header">لطفاً دسته بندی وجود ندارد</p>
28   </div>';
29 } else { ?>
30
31
32 <?php for ($i=1; $i <= count($arr_all); ) { ?>
33
34   <div class="row">
35
36     <?php for ($j=1; $j <= 3; $j++) { ?>
37
38       <?php if ($i+$j-2 == count($arr_all)) {
39         break;
40       } ?>
41
42       <div class="col s12 m4">
43         <div class="card">
44           <div class="card-image waves-effect waves-block waves-light">
```

Figure 6-3, Code of the categories Page

In this page, we want to view the categorization of foods. For example, categories like American cuisine or Mediterranean cuisine. To achieve this, we have used a for loop. The for loop in PHP is used to iterate over a specified number of times. There are two key commands in using the for loop in PHP to control the repetition of internal commands within the loop. One is the "break" command, which can be used to prevent an infinite loop from occurring. This command is used to interrupt and break the normal execution of a statement within a loop under certain conditions. Another command is "continue." With this command, it's possible to skip one execution step in the loop and jump to the next step.

4) logout.php

The screenshot shows the Adobe Dreamweaver interface. The title bar says "Dw" and the menu bar includes File, Edit, View, Insert, Tools, Find, Site, Window, Help. The tabs at the top show "index.php" and "logout.php". The main code editor window contains the following PHP code:

```
<?php
session_start();
session_destroy();
session_start();
header('location: index.php');
```

A tooltip above the code states: "Dynamically-related files cannot be discovered because there is no site definition for this document. Setup". The right side of the interface shows the "Files" panel with a tree view of the "SaraSite" folder containing various files like "eula+sci.jpg", "Wallpaper.jpg", etc. The "DOM" panel at the bottom right displays the DOM structure of the HTML documents.

Figure 7-3, the code for the logout page.

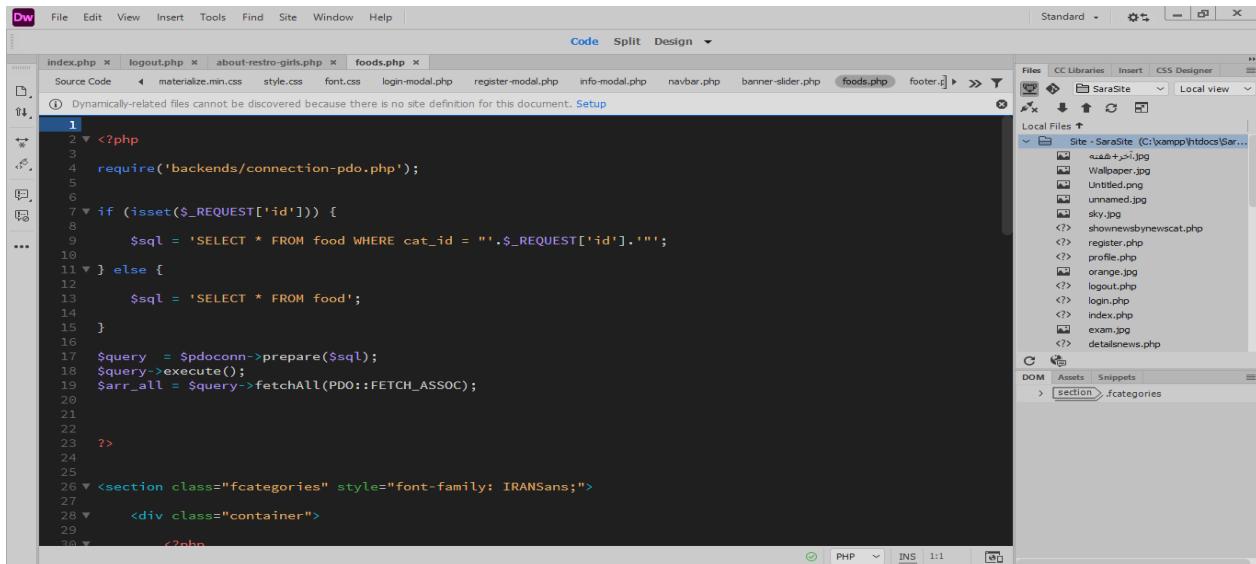
Perhaps the logout page is the simplest page in terms of code. This page is composed of a session and a header. When working with a program, you open it, do some tasks, and then close it. This is very similar to a session. The computer knows who you are, knows when you opened the program, and knows when you closed it. However, on the internet, there is a problem: the web server doesn't know who or what you are because the HTTP address doesn't retain the state.

session_start: A session begins with the session_start function.

session_destroy: This function removes all information in the session along with the session file from the server. It does not require any arguments. If you simply close your browser without clicking the logout link, your session will remain, and someone with your address can still access it.

header: By using the header command in PHP, we can redirect the user to a new page (the user's browser opens a new page).

5) foods.php



The screenshot shows the Adobe Dreamweaver interface with the 'Code' tab selected. The main editor window displays the PHP code for the 'foods.php' page. The code includes database connection setup, conditional logic for retrieving food items based on a category ID from the \$_REQUEST superglobal variable, and the execution of SQL queries using PDO. The right panel shows the 'Local Files' panel with a tree view of the website's file structure under 'Site - SaraSite'. The bottom status bar indicates the code is in PHP mode.

```
1 <?php
2
3
4 require('backends/connection-pdo.php');
5
6
7 if (isset($_REQUEST['id'])) {
8
9     $sql = 'SELECT * FROM food WHERE cat_id = "'.$_REQUEST['id'].'."'";
10
11 } else {
12
13     $sql = 'SELECT * FROM food';
14
15 }
16
17 $query = $pdoconn->prepare($sql);
18 $query->execute();
19 $arr_all = $query->fetchAll(PDO::FETCH_ASSOC);
20
21
22 ?>
23
24
25 <section class="fcategories" style="font-family: IRANSans;">
26
27     <div class="container">
28
29         <?php
```

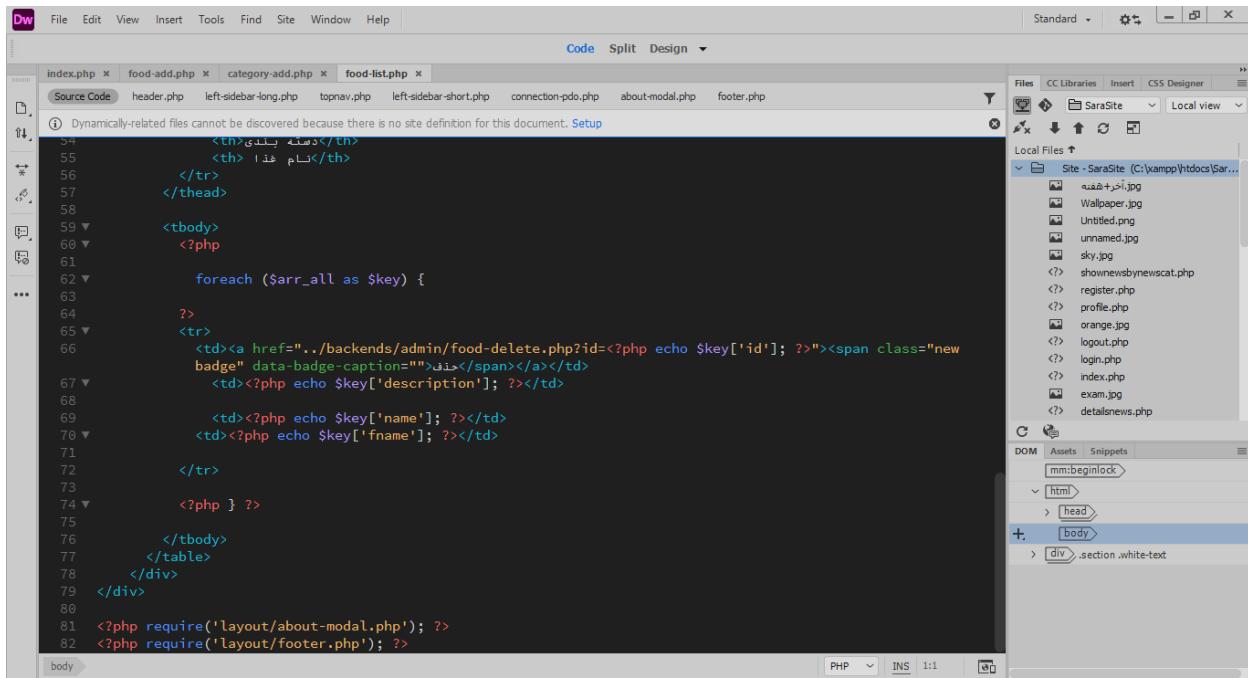
Figure 8-3, the code for the foods page.

In this page, we want to view the foods along with their descriptions and related information. After logging into their profile, users can order their desired food from this page. The `$_REQUEST` variable is a superglobal variable that contains the contents of three other superglobal variables:

- `$_GET`
- `$_POST`
- `$_COOKIE`

In a PHP page, to access the value of a variable from both GET and POST methods, it is sufficient to write code for the `$_REQUEST` superglobal variable, and there is no need for separate coding for the superglobal variables `$_GET` and `$_POST`.

5-1) food-list.php



The screenshot shows the Adobe Dreamweaver interface with the 'Code' tab selected. The main window displays the PHP code for the 'food-list.php' page. The code includes HTML structure for a table, a foreach loop to iterate through an array '\$arr_all', and conditional logic for deleting items via links. The right panel shows the 'Local Files' panel with various site files listed, and the bottom right shows the DOM tree.

```
<?php require('layout/about-modal.php'); ?>
<?php require('layout/footer.php'); ?>
```

```
54     <th>نامه</th>
55     <th>پسته</th>
56   </tr>
57 </thead>
58
59 <tbody>
60
61   <foreach ($arr_all as $key) {
62
63     ?>
64     <tr>
65       <td><a href="../backends/admin/food-delete.php?id=<?php echo $key['id']; ?>"><span class="new badge" data-badge-caption="<?php echo $key['category']; ?></span></a></td>
66         <td><?php echo $key['description']; ?></td>
67
68         <td><?php echo $key['name']; ?></td>
69         <td><?php echo $key['fname']; ?></td>
70
71     </tr>
72
73   <?php } ?>
74
75 </tbody>
76 </table>
77 </div>
78 </div>
79 </div>
80
81 <?php require('layout/about-modal.php'); ?>
82 <?php require('layout/footer.php'); ?>
```

Figure 9-3, the code for the food-list page

In this page, besides viewing the menu of foods, their names, categorization, and descriptions, we want to perform deletion operations. Therefore, we use the `<a>` tag and set its `href` attribute to the name of our desired page, which, in this example, is `food-delete.php`. When the delete link is clicked, the value of the food category number, i.e., `id`, is placed inside a variable with a chosen name (here, we consider it as `id`), and it is sent to the `food-delete.php` page using the GET method through the URL in the address bar. The question mark placed after the page address indicates the usage of the GET method.

The screenshot shows the Adobe Dreamweaver interface with the 'Code' tab selected. The main area displays the PHP code for the 'food-list.php' page. The code includes session handling, CSS styling for sections, and an HTML table for food items. The right panel shows the 'Local Files' panel with various site files listed. The bottom status bar shows the current file path and line number.

```
<?php
if (isset($_SESSION['msg'])) {
    echo '<div class="section center" style="margin: 5px 35px;"><div class="row" style="background-color: #f0f0f0; padding: 10px; border-radius: 5px; text-align: center;">
        <div class="col s12">
            <h6>' . $_SESSION['msg'] . '</h6>
        </div>
    </div></div>';
    unset($_SESSION['msg']);
}
?>

<div class="section left" style="padding: 15px 25px;">
    <a href="food-add.php" class="waves-effect waves-light btn">کردن فایل جدید</a>
</div>

<div class="section center" style="padding: 20px; font-family: IRANSans;">
    <table class="centered responsive-table">
        <thead>
            <tr>
                <th>عنوان</th>
                <th>تصویر</th>
                <th>دسته بندی</th>
                <th>فایل</th>
            </tr>
        </thead>
```

Figure 1-9-3, the code for the food-list page

Continuing with this page, we are now going to perform the operation of adding a new food. Therefore, we use the `<a>` tag and set its `href` attribute to the name of our desired page, which, in this case, is `food-add.php`.

5-2) food-delete.php

The screenshot shows the Adobe Dreamweaver interface with the 'Code' tab selected. The main window displays the PHP code for the 'food-delete.php' file. The code handles the deletion of a record from a database table named 'food'. It first checks if the ID is set via the GET method. If not, it redirects to the 'food-list.php' page. If the ID is set, it constructs a DELETE query, prepares it with PDO, and executes it. After execution, it sets a session message indicating the item was deleted and then redirects to the 'food-list.php' page. The right panel shows the local file structure of the website, which includes various PHP files like index.php, category-add.php, and food-list.php, along with some image files.

```
20
21 if (!isset($_REQUEST['id'])) {
22
23     $_SESSION['msg'] = 'فیلر ثبت نامعتبر است';
24
25     header('location: ../../admin/food-list.php');
26
27     exit();
28 }
29
30 $id = $_REQUEST['id'];
31
32
33 $sql = "DELETE FROM food WHERE id = ?";
34 $query = $pdoconn->prepare($sql);
35 if ($query->execute([$id])) {
36
37     $_SESSION['msg'] = 'آیتم مورد نظر حذف شد';
38
39     header('location: ../../admin/food-list.php');
40 }
41 else {
42
43     $_SESSION['msg'] = 'اچکالی در سرور بوجود آمده لطفاً بعد از کار را انجام دهید';
44
45     header('location: ../../admin/food-list.php');
46 }
47
48
49 }
```

Figure 10-3, the code for the food-list page

In the food-delete.php page, the first thing that needs to be done is to retrieve and store the value of the ID sent via the GET method. Now, we should use this variable in the WHERE part of the DELETE statement to ensure that only the record with that ID is deleted. If there are no errors during the execution of the query, we can use the header command to redirect to the specified location. In this example, we will redirect to the food-list.php page, and since this page is reloaded, the deleted record won't be visible anymore.

5-3) food-add.php

The screenshot shows the Adobe Dreamweaver interface with the 'Code' tab selected. The main panel displays the PHP code for the 'food-add.php' page. The code handles form submission, connects to a database using PDO, and inserts new food items into a database table. It also sets session variables and uses header redirection to return to the food-list.php page after a successful addition. The right panel shows the site's file structure under 'Local Files'.

```
① Dynamically-related files cannot be discovered because there is no site definition for this document. Setup
22 if (!isset($_POST['name']) || !isset($_POST['desc'])) {
23     $SESSION['msg'] = 'لطفاً معرفاً صنفه و تاره کنید POST کلیدهای متغیر';
24     header('location: ../../admin/food-list.php');
25     exit();
26 }
27
28 $name = $_POST['name'];
29 $desc = $_POST['desc'];
30 $category = $_POST['category'];
31
32 $sql = "INSERT INTO food(cat_id, fname, description) VALUES(?, ?, ?)";
33 $query = $pdoconn->prepare($sql);
34 if ($query->execute([$category, $name, $desc])) {
35     $SESSION['msg'] = 'آشپزی جدید اضافه شد';
36     header('location: ../../admin/food-list.php');
37 } else {
38     $SESSION['msg'] = 'مشکلی در سرور بوجود آمده است لطفاً بعد امتحان کنید';
39     header('location: ../../admin/food-list.php');
40 }
```

Figure 11-3, the code for the food-add page

In the food-add.php page, after connecting to the database, we need to use the INSERT command to add the desired record to the list of foods. If there are no errors during the execution of the query, we can use the header command to redirect to the specified location. In this example, we will redirect to the food-list.php page, and since this page is reloaded, the added record will be visible. Then, through a session and a message, we inform the site admin that "a new food has been added.

Admin Table

The screenshot shows the phpMyAdmin interface for the 'admin' table in the 'onlinerestoran' database. The table has four columns: id, name, email, and password. The 'id' column is defined as int(11) with AUTO_INCREMENT, 'name' as varchar(250), 'email' as varchar(50), and 'password' as varchar(250). An index named 'PRIMARY' is defined on the 'id' column.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)	utf8_persian_ci	No	None		AUTO_INCREMENT		Change Drop More
2	name	varchar(250)	utf8_persian_ci	No	None				Change Drop More
3	email	varchar(50)	utf8_persian_ci	No	None				Change Drop More
4	password	varchar(250)	utf8_persian_ci	No	None				Change Drop More

Figure 12-3, Admin table

Food Categories Table

The screenshot shows the phpMyAdmin interface for the 'categories' table in the 'onlinerestoran' database. The table has four columns: id, name, short_desc, and long_desc. The 'id' column is defined as int(11) with AUTO_INCREMENT, 'name' as varchar(250), 'short_desc' as varchar(250), and 'long_desc' as varchar(500). An index named 'PRIMARY' is defined on the 'id' column.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)	utf8_persian_ci	No	None		AUTO_INCREMENT		Change Drop More
2	name	varchar(250)	utf8_persian_ci	No	None				Change Drop More
3	short_desc	varchar(250)	utf8_persian_ci	No	None				Change Drop More
4	long_desc	varchar(500)	utf8_persian_ci	No	None				Change Drop More

Figure 13-3, Food categories table

Foods Table

The screenshot shows the phpMyAdmin interface for the 'onlinerestoran' database. The left sidebar lists various databases and tables, with 'food' selected. The main panel displays the 'Table structure' for the 'food' table. The table has four columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	Change Drop More	
2	cat_id	int(10)			No	None		Change Drop More	
3	fname	varchar(50)	utf8_persian_ci		No	None		Change Drop More	
4	description	varchar(250)	utf8_persian_ci		No	None		Change Drop More	

Below the table structure, there are sections for 'Indexes' and 'Partitions'. The 'Indexes' section shows one primary key index:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	4	A	No	

The 'Partitions' section is currently empty.

Figure 14-3, Foods table

Comments Table

The screenshot shows the phpMyAdmin interface for the 'onlinerestoran' database. The left sidebar lists various databases and tables, with 'message' selected. The main panel displays the 'Table structure' for the 'message' table. The table has five columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	Change Drop More	
2	name	varchar(30)	utf8_persian_ci		Yes	NULL		Change Drop More	
3	email	varchar(100)	utf8_persian_ci		Yes	NULL		Change Drop More	
4	message	text	utf8_persian_ci		Yes	NULL		Change Drop More	
5	date	timestamp			No	current_timestamp()		Change Drop More	

Below the table structure, there are sections for 'Indexes' and 'Partitions'. The 'Indexes' section shows one primary key index:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	5	A	No	

The 'Partitions' section is currently empty.

Figure 15-3, comments table

Orders Table

The screenshot shows the phpMyAdmin interface for the 'orders' table in the 'onlinerestoran' database. The left sidebar lists various databases and tables. The main area displays the table structure with 6 columns: id, order_id, user_id, food_id, user_name, and timestamp. A primary key index is defined on the 'id' column.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)	utf8_persian_ci	No	None		AUTO_INCREMENT		Change Drop More
2	order_id	varchar(20)	utf8_persian_ci	No	None				Change Drop More
3	user_id	varchar(10)	utf8_persian_ci	No	None				Change Drop More
4	food_id	varchar(10)	utf8_persian_ci	No	None				Change Drop More
5	user_name	varchar(100)	utf8_persian_ci	No	None				Change Drop More
6	timestamp	varchar(50)	utf8_persian_ci	No	None				Change Drop More

Figure 16-3, orders table

Users Table

The screenshot shows the phpMyAdmin interface for the 'users' table in the 'onlinerestoran' database. The left sidebar lists various databases and tables. The main area displays the table structure with 7 columns: id, name, email, password, timestamp, address, and number. A primary key index is defined on the 'id' column.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)	utf8_persian_ci	No	None		AUTO_INCREMENT		Change Drop More
2	name	varchar(50)	utf8_persian_ci	Yes	NULL				Change Drop More
3	email	varchar(50)	utf8_persian_ci	Yes	NULL				Change Drop More
4	password	varchar(100)	utf8_persian_ci	Yes	NULL				Change Drop More
5	timestamp	varchar(100)	utf8_persian_ci	Yes	NULL				Change Drop More
6	address	varchar(200)	utf8_persian_ci	No	None				Change Drop More
7	number	int(20)		No	None				Change Drop More

Figure 17-3, users table

Chapter Four: Examination of Various Website Pages



Figure 1-4: User Login Steps on the Website

The screenshot shows a web browser window for 'User Panel' on 'localhost:8080'. The top navigation bar includes links for 'ورود' (Login), 'عضویت' (Registration), 'بخش مدیریت' (Management Section), 'تمامن با ما' (About Us), 'نظرات' (Reviews), 'درباره ما' (About Us), 'غذاها' (Food), 'دسته بندی ها' (Categories), and 'صفحه اصلی' (Main Page). The main content area features a banner with Persian text: 'به آنلاین فود خوش آمدید' (Welcome to Online Food) and 'بهترین کیفیت با رستوران آنلاین' (Best quality with online restaurant). Below the banner, there is a promotional message: 'ما کیفیت را تحویل می دهیم. اول امتحان کنید بعدا از ما خرید کنید!' (We deliver quality. Try us first, then buy from us!). At the bottom, there is a large headline: 'کادر مجرب رستوران آماده سرویس دهی' (Experienced staff ready to serve).

Figure 2-1-4: Website Start

User enters the site. If they have not registered before, they click on the "Sign Up" button and proceed to register. Otherwise, they click on the "Login" button to enter the site.

Figure 3-1-4: Sign Up

The user proceeds to register. The first and last name must be in Persian, and the email must be valid. All fields marked with an asterisk (*) must be filled. The password and its confirmation must match. Finally, by clicking the "Register" button, the user enters the login page.

Figure 4-1-4: Sign Up

The user enters the website by entering a valid email and password and clicking the login button. Then, they are greeted with a welcome message on the main page.

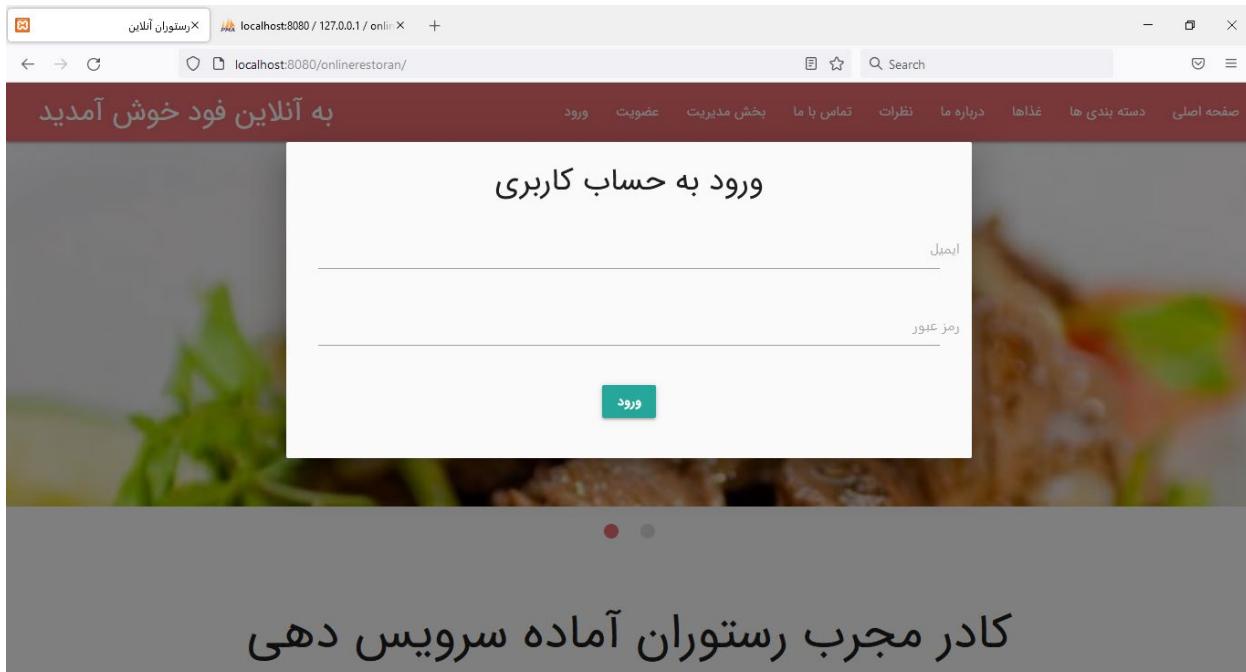


Figure 5-1-4: Log in

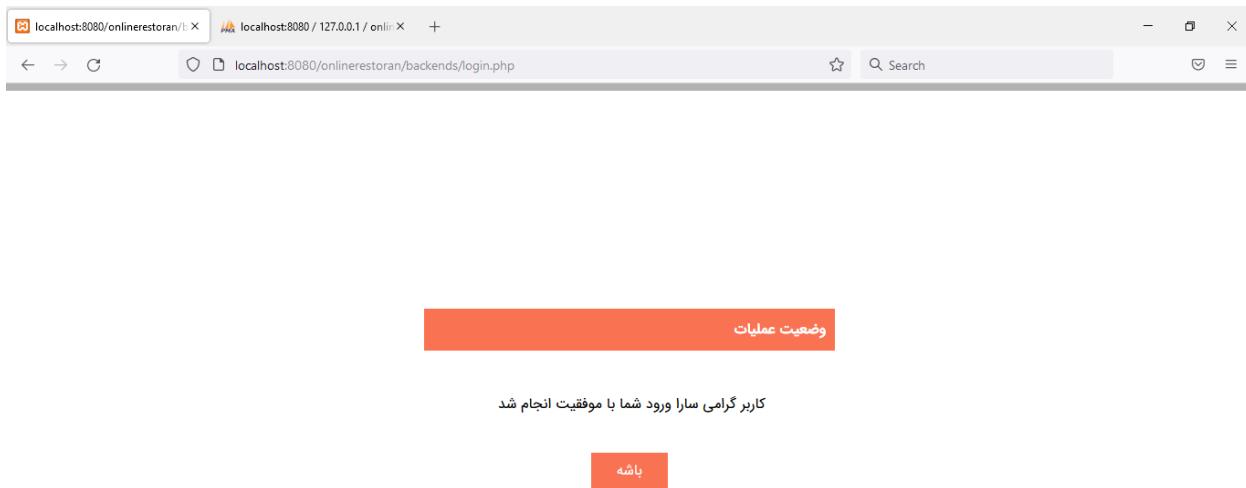


Figure 6-1-4: Log in

The user logs in and can navigate to different sections of the website by clicking on various options in the site header, as shown in the image. Additionally, the user can log out of their profile by selecting the "Logout" option and return to the main page of the site.

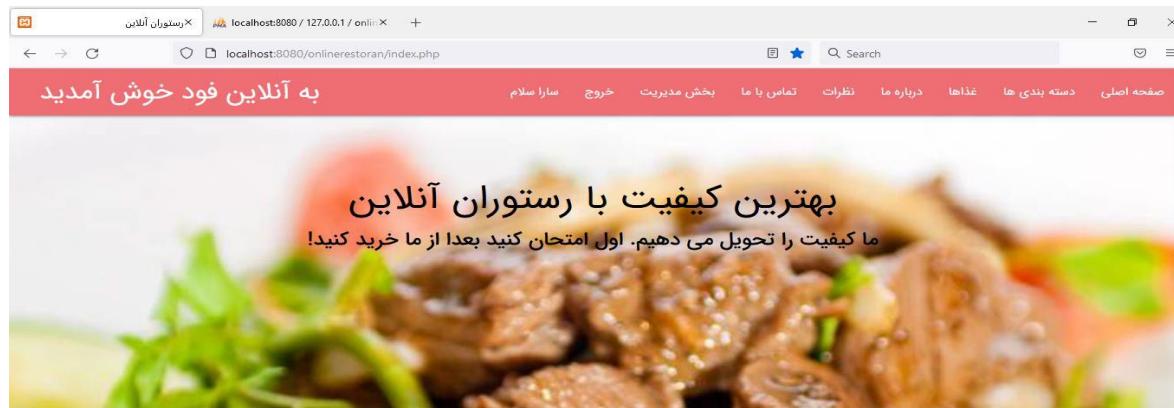


Figure 5-1-4

5-1-4 User profile

The user, by clicking on the "More" option, enters the "Food Categories" page.

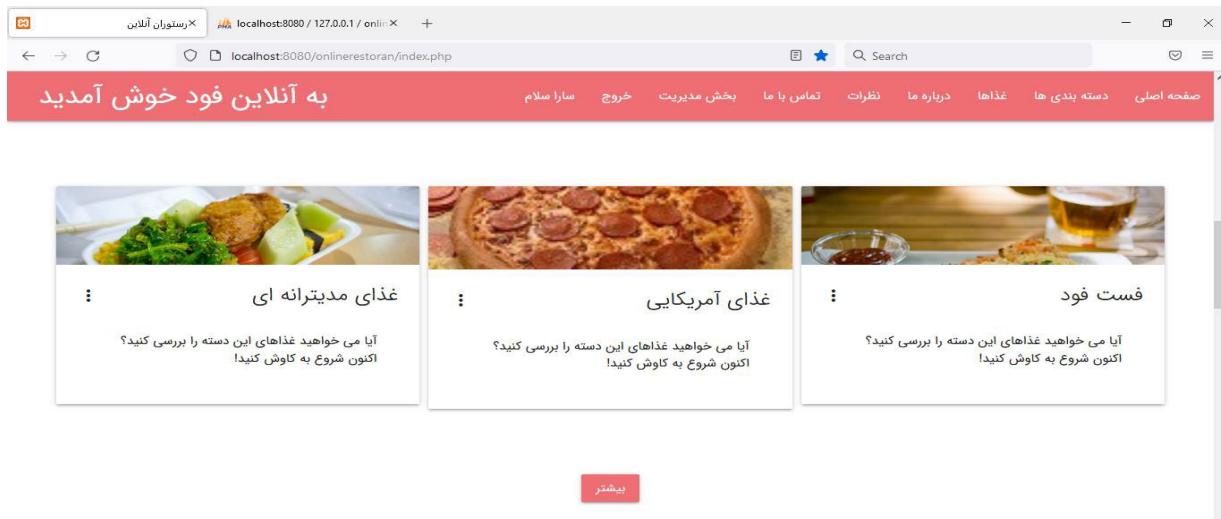


Figure 2-6-1-4 Site Content Description

User clicks on the "Read more" option to enter the "About Us" page.

The screenshot shows a web browser window for 'Restoran آنلاین' at 'localhost:8080'. The main title is 'به آنلاین فود خوش آمدید'. The page content discusses the quality of food and service, mentioning various dishes and their descriptions. A large image of a meal is displayed at the top. A blue button labeled 'بیشتر بخوانید' (Read more) is visible at the bottom of the text block.

غذاخوری‌ها از دیدگاه نوع مشتریان، بهای خوراک‌ها و کیفیت خدمات ارائه شده، اغلب به دو دسته بخش می‌شوند، دسته نخست، غذاخوری‌های بی‌تكلف و ارزان قیمت‌اند که بیشتر مشتریانشان از باشندگان همان محل مستنده است. در همان محدوده کار می‌کنند. میهمانان مقدم به رعایت آداب و رسوم ویژه‌ای نیستند و بیشتر با لباس‌های معمولی و غیررسمی در غذاخوری‌های حاضر شوند. دسته دوم غذاخوری‌های گران‌قیمت هستند. در این گونه غذاخوری‌ها، جهای خوراک‌ها و خدمات بالاست و بسته به عرف و فرهنگ جوامع گوناگون، میهمانان با پوشش رسمی یا نیمه‌رسمی حضور می‌یابند. ارایه غذای رسانی در رستوران با همکاری همه کارمندان رستوران انجام می‌شود. چیز ادوات غذاخوری، ارایه نو و دریافت سفارش از میهمانان و ارایه غذا به اینسان، تحول صورت‌حساب، دریافت وجه و در پایان مرتب‌ساختن و پیراشتن میز از ظایف پادوهاس است. پخت غذا نیز از ظایف آشیزها بهشمار می‌آید. غذاخوری از خوراک‌ها را ارائه کنند، مانند کوئه‌ای از غذاخوری است که در آن خود میهمانان از خوبیشان بذریعی می‌کنند. همچنین، غذاخوری‌ها ممکن است گونه ویژه‌ای را ارائه کنند، مانند خوراک‌های دریابی، خوراک‌های گیاهی با خوراک‌های ویرژ ملت‌ها و اقوام گوناگون. مانند رستوران ایرانی، چینی، لیبانی و ...).

بیشتر بخوانید

Figure 3-6-1-4 Site Content Description

By clicking on the "Customer Reviews," the user can read the reviews. Always, the restaurant information, including the phone number and address, is visible.

The screenshot shows a web browser window for 'Restoran آنلاین' at 'localhost:8080'. The main title is 'به آنلاین فود خوش آمدید'. The page content displays customer reviews with icons for each review. The reviews are in Persian and discuss the quality of service and food.

متفاوت در سرویس دهی
تحویل رایگان
کیفیت عالی

نظر مشتریان ما

درباره ما
مکان: مشهد - رستوران آنلاین
تلفن: 5130000000
سرویس همه روزه در طول ایام هفته

شبکه های اجتماعی
Facebook
Instagram
Twitter
Whatsapp

Figure 4-6-1-4 Site Content Description

By clicking on the "Categories" option, the user enters the relevant page and can view the categorization of various foods. Also, by clicking on the three dots next to the image, a window opens, allowing the user to read more information about this categorization.

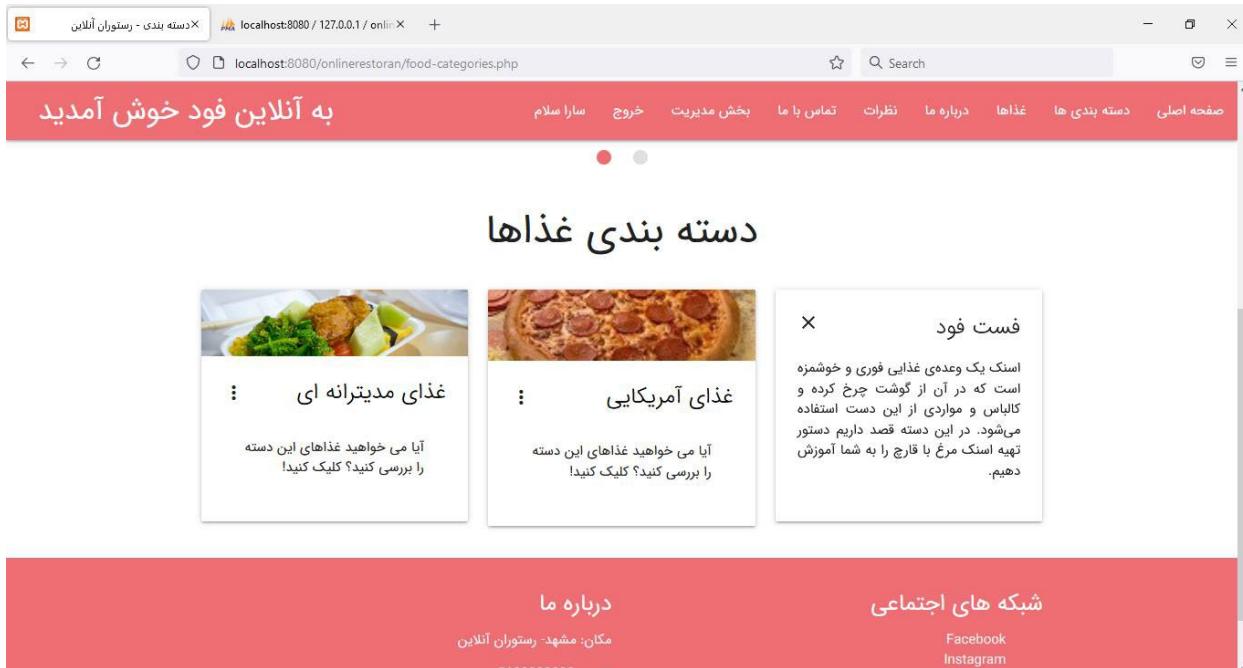


Figure 7-1-4 Description of Website Content

By clicking on the "Foods" option, the user enters the respective page and can view the list of foods. Then, to place an order, it is sufficient to click on the "Order Now" option. It is worth mentioning that if the user has not logged in through the login page initially, they cannot order food and will encounter an error message.

By clicking on the "Comments" option, the user enters the respective page and, through the form provided, can send their comments to the site admin. This message is then stored in the database related to comments. It is essential to fill in the starred fields. Also, by clicking on the "Exit" button, the user returns to the main page. If the message is sent successfully without any issues, a confirmation message is displayed to the user.



Figure 8-1-4 Description of Website Content

If the message is successfully submitted, you will be faced with the following message: "Your comment has been successfully submitted. Thank you for your feedback!"

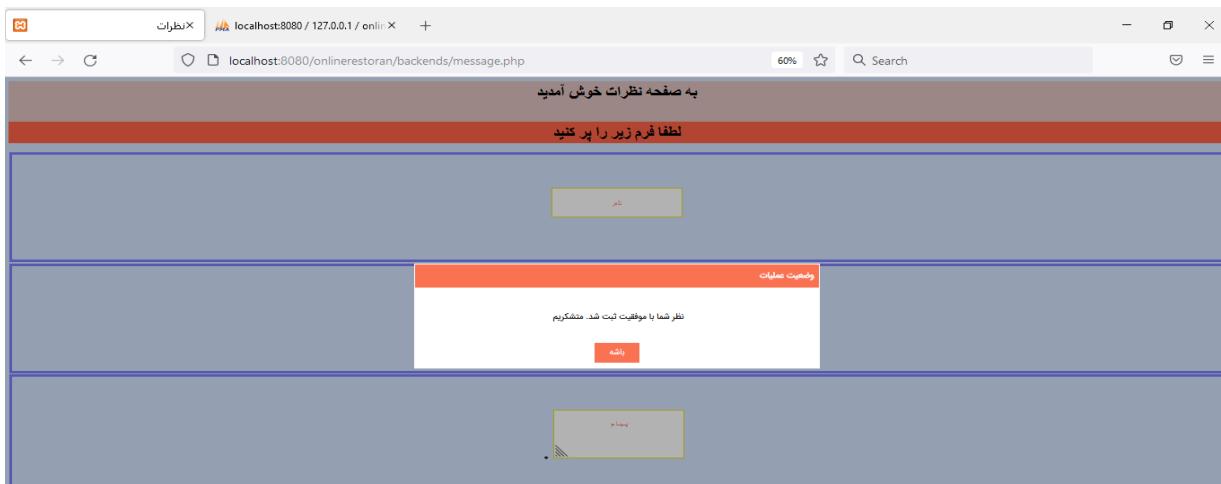


Figure 9-1-4 Description of Website Content

By clicking on the "Contact Us" option, the user will enter the relevant page and can view information about the restaurant. Additionally, links through which the user can connect with the restaurant in the online space will also be provided.

The screenshot shows a web browser window for a website titled 'تماس با ما - رستوران آنلاین' (Contact Us - Online Restaurant). The URL in the address bar is 'localhost:8080/online_restoran/contact.php'. The page has a red header bar with the text 'به آنلاین فود خوش آمدید' (Welcome to Online Food). The main content area has a light blue background and features two columns. The left column is titled 'درباره ما' (About Us) and contains the following information:
مکان: مشهد
تلفن: 5130000000
دارای سرویس رایگان
شماره تماس با مدیر: 0214455669988
سرویس آنلاین در تمامی روزهای هفتگی
شنبه تا چهارمینه

The right column is titled 'شبکه های اجتماعی' (Social Networks) and lists the following links:
Facebook
Instagram
Twitter
Whatsapp

Figure 10-1-4 Description of Website Content

Admin Panel

The password and username of the administrator have been manually entered into the database, and only the administrator is aware of them. I am completing the login form. The manager can navigate to different sections of the site and perform the desired operations.



Figure 1-2-4: Admin Login Page

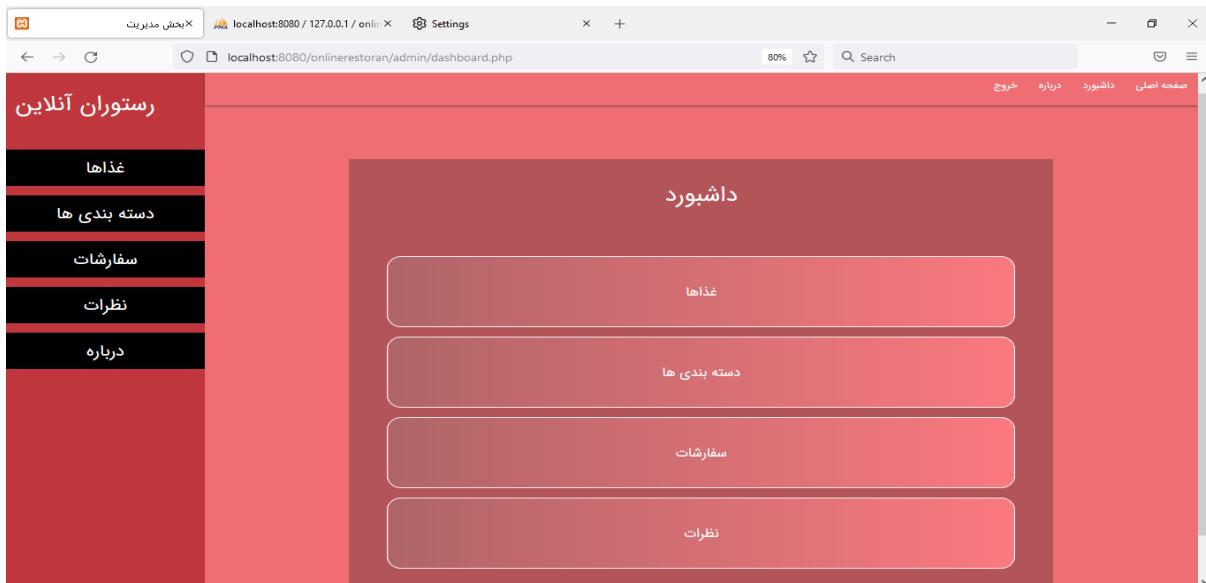


Figure 2-2-4: Admin Login Page

The manager can view the food menu and, if necessary, add or remove a dish by clicking on the relevant operation.

By clicking on the "Add New Food" option, the manager enters the corresponding page and can add a new item to the menu.

نام غذا	دسته بندی	توضیف	عملیات
کیسهه ی مرغ	غذایی مدیرانه ای	کیسهه ی مرغ یکی از خوشمزه ترین غذا های مدیرانه ای هست. مواد تشکیل دهنده: پیاز سیر مرغ برجسته فرنگی چل دارچین آب پرتقال زعفران رز و چوبه رب	حذف
پیترو یوروی	غذایی مدیرانه ای	پیترو یک نوع سالمی امریکایی است که از گوشت گاو پخته شده با چاهن، پاپریکا یا دیگر ظرفی تهیه می شود. مواد تشکیل دهنده: رنگ زیتون سیر پیپر پیپریار مرزن رزده شده سوسیس بیرونی تخم خشک فلیل قرمز	حذف
کباب ترکی(گوشت)	غذایی مدیرانه ای	این ساندویچ به طور کلی سالاد را میزباند که شامل گوجه فرنگی، کاهو، کلم، پیاز انواع مختلف مسوس باشد. مواد تشکیل دهنده: گوشت قرمز قارچ فلفل دلمه ای سیر پیاز اونوش	حذف
ساندویچ قبله مرغ	غذایی آمریکایی	یک نوع در خوردن ساندویچ است که علاوه بر لذت بودن آن، نسبت به دیگر قسم فودها سالم تر است. مواد تشکیل دهنده: فیله مرغ ماست آب لیمو توپش روغن زیتون مس همیرگ	حذف

Figure 3-2-4: Admin Login Page

اضافه کردن غذای جدید

نام غذا	انتخاب دسته
غذایی مدیرانه ای	<input type="button" value="غذایی، مدیرانه ای"/>
توضیف	
<input type="button" value="انصراف"/> <input type="button" value="اضافه کردن"/>	

Figure 4-2-4: Admin Login Page

Additionally, by selecting the "Delete" option, as you can see, the food has been removed from the food menu.

Similar to the food menu page, in this section, you can perform delete and add operations.

The screenshot shows a web application for managing a restaurant's food menu. The left sidebar has navigation links: 'Food Categories' (غذاها), 'Food Bundles' (دسته بندی ها), 'Orders' (سفارشات), 'Reviews' (نظرات), and 'Inventory' (درباره). The main content area is titled 'Food Categories' (منوی غذا ها) and displays a table of food items. A red banner at the top says 'Item removed from the menu' (آیتم مواد نظر حذف شد). The table columns are: Name (نام غذا), Description (توضیحات), Category (دسته بندی), and Status (وضعیت). One row is highlighted in red, indicating it has been deleted. The status column shows a 'Deleted' button (حذف).

نام غذا	توضیحات	دسته بندی	وضعیت
کیسه های مرغ یکی از خوشمزه ترین غذا های مدیرانه ای هست. مواد تشکیل دهنده: پیاز سبز برنج تره فرنگی	غذاهای مرغ	مدیرانه ای	حذف
پیروزی یک نوع سلاس آمریکایی است که از گوشت گاو و یخچه شده با چاشنی پارزکا با دیگر گلظل چیل تهیه می شود. مواد تشکیل دهنده: رون زیتون سیر پیبر پیمان رنده شده سوسیس پیروزی تخم خشک گلظل قرمز	غذاهای پیروزی	مدیرانه ای	حذف
این ساندویچ به طور کلی حاوی سلاس با سیریجات است که شامل گوچه فرنگی، کاهو، کلم، پیاز آوجیشن	غذاهای ساندویچ	مدیرانه ای	حذف
یک تنوع در خودن ساندویچ است که علاوه بر لذیذ بودن آن، نسبت به دیگر فست فودها سالم تر است. مواد تشکیل دهنده: فله مرغ ماسات اب یمیو توش زیتون سس همراه	ساندویچ فله مرغ	آمریکایی	حذف

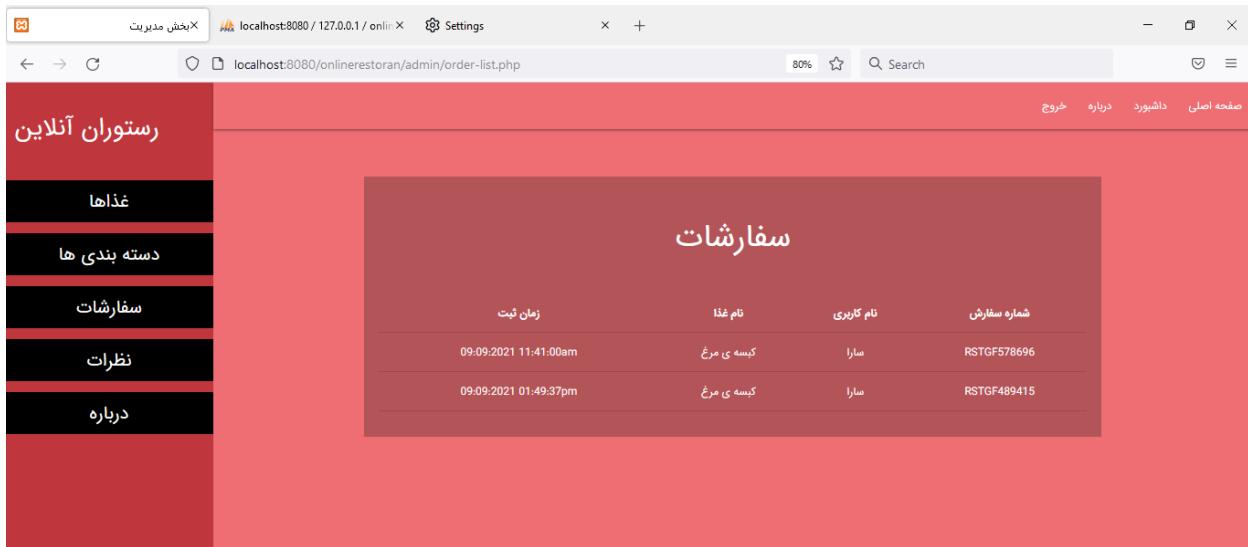
Figure 5-2-4: Admin Login Page

The screenshot shows a web application for managing categories. The left sidebar has navigation links: 'Food Categories' (غذاها), 'Food Bundles' (دسته بندی ها), 'Orders' (سفارشات), 'Reviews' (نظرات), and 'Inventory' (درباره). The main content area is titled 'Food Bundles' (دسته بندی ها) and displays a table of categories. A red banner at the top says 'Add new category' (اضافه کردن دسته بندی جدید). The table columns are: Name (نام دسته)، Description (توضیح کوتاه)، Status (وضعیت)، and Category (دسته بندی). One row is highlighted in red, indicating it has been deleted. The status column shows a 'Deleted' button (حذف).

نام دسته	توضیح کوتاه	دسته بندی	وضعیت
آیا می خواهد غذاهای این دسته را بروز نماید؟ کلیک!	این آشپزی خوشمزه و لذیذ و سلامت قدمتی به درازی مصر باستان دارد و شواهدی دار بر وجود این آشپزی در قرون های ۳ و ۴ بعد از میلاد وجود دارد. با این حال، مفهوم و اصطلاح آشپزی مدیرانه ای به تکلی که ما این شناسنامه بر پندهان ندارد	غذاهای این دسته	حذف
آیا می خواهد غذاهای این دسته را بروز نماید؟ کلیک!	هات داگ ها اختصاراً معروف ترین غذاهای خیابانی آمریکایی هستند. سوسیس به سبک آلمانی، که پخار پیز یا کبابی می شوند و به عنوان فرنگری فود نام فرازه داده می شوند	غذاهای آمریکایی	حذف
آیا می خواهد غذاهای این دسته را بروز نماید؟ کلیک!	اسکی ری و یونده غذاهای فوری و خوشمزه است که در آن از گوشت چرخ کرده و کالباس و موادی از این دست استفاده می شوند. در این دسته قصد داریم دستور تهیه اسنک مرغ با قارچ را به شما آموخته دهیم	غذاهای این دسته	حذف

Figure 6-2-4: Admin Login Page

In this page, the manager can view the registered orders and track them if necessary using the order numbers.

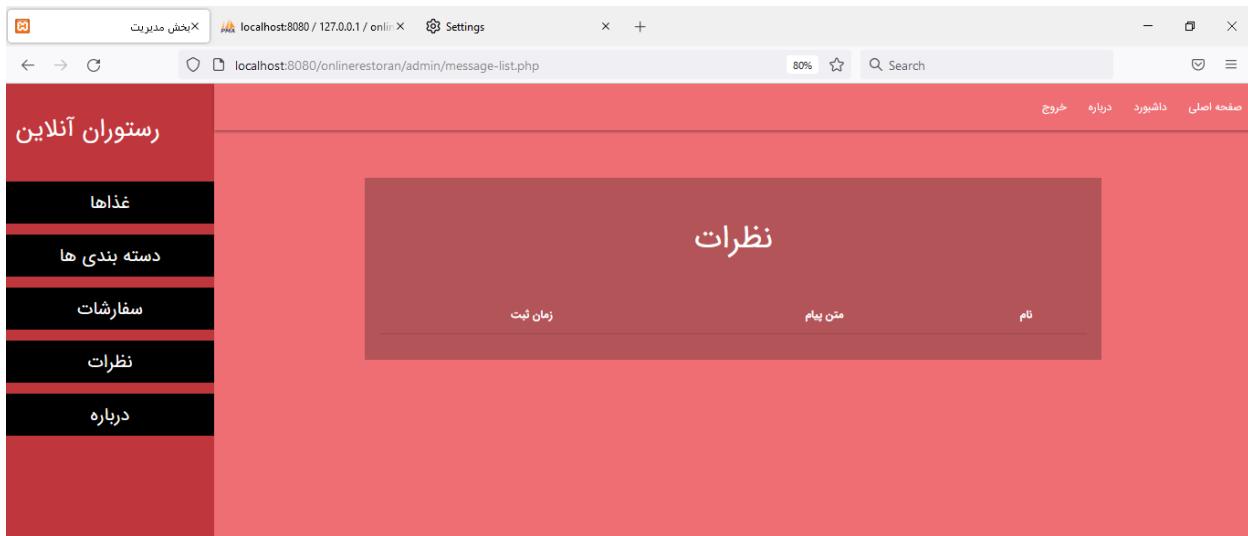


A screenshot of a web browser showing the 'order-list.php' page. The left sidebar has menu items: 'غذاها' (Food), 'دسته بندی ها' (Categories), 'سفارشات' (Orders), 'نظرات' (Comments), and 'درباره' (About). The main content area is titled 'سفارشات' (Orders) and displays a table with two rows of order information:

زمان ثبت	نام غذا	نام کاربری	شماره سفارش
09-09-2021 11:41:00am	کبکه ی مرغ	سارا	RSTGF578696
09-09-2021 01:49:37pm	کبکه ی مرغ	سارا	RSTGF489415

Figure 7-2-4: Admin Login Page

The manager can also view the submitted comments in this section, although currently, there are no comments.



A screenshot of a web browser showing the 'message-list.php' page. The left sidebar has the same menu items as the previous page. The main content area is titled 'نظرات' (Comments) and displays a table with three columns:

زمان ثبت	متن پیام	نام

Figure 8-2-4: Admin Login Page

The footer of the management panel indicates that the site's rights belong to the manager.

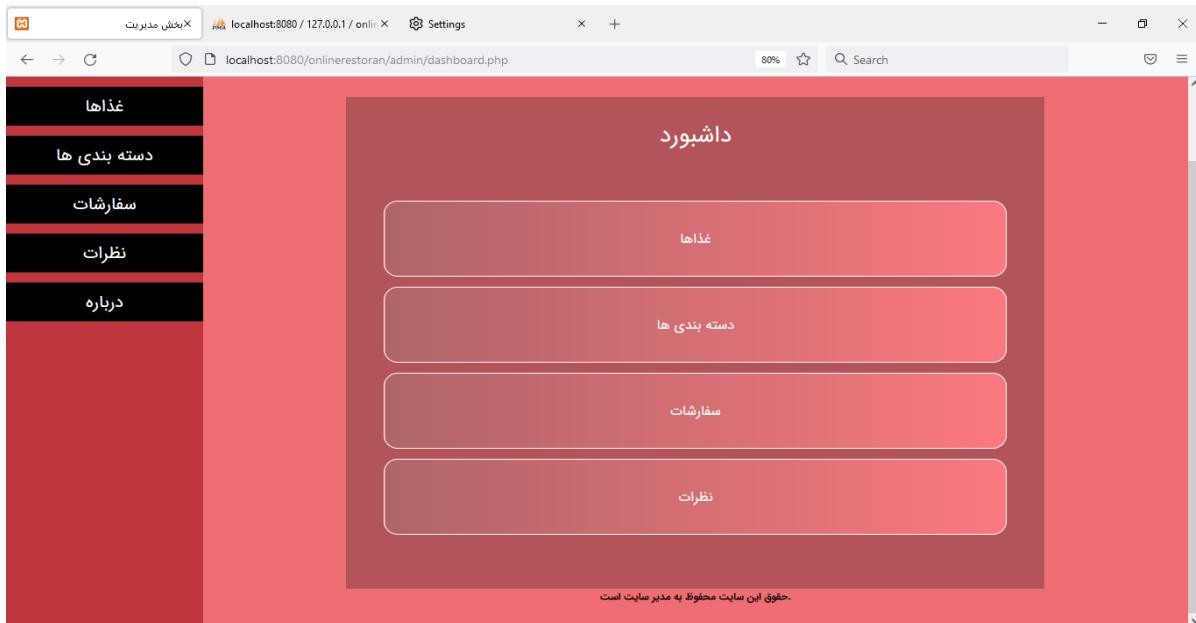


Figure 9-2-4: Admin Login Page

Finally, the manager can log out of the management panel by selecting the "Logout" option, which is confirmed with the above message. Additionally, by selecting the "Homepage" option, the manager can return to the site's main page.

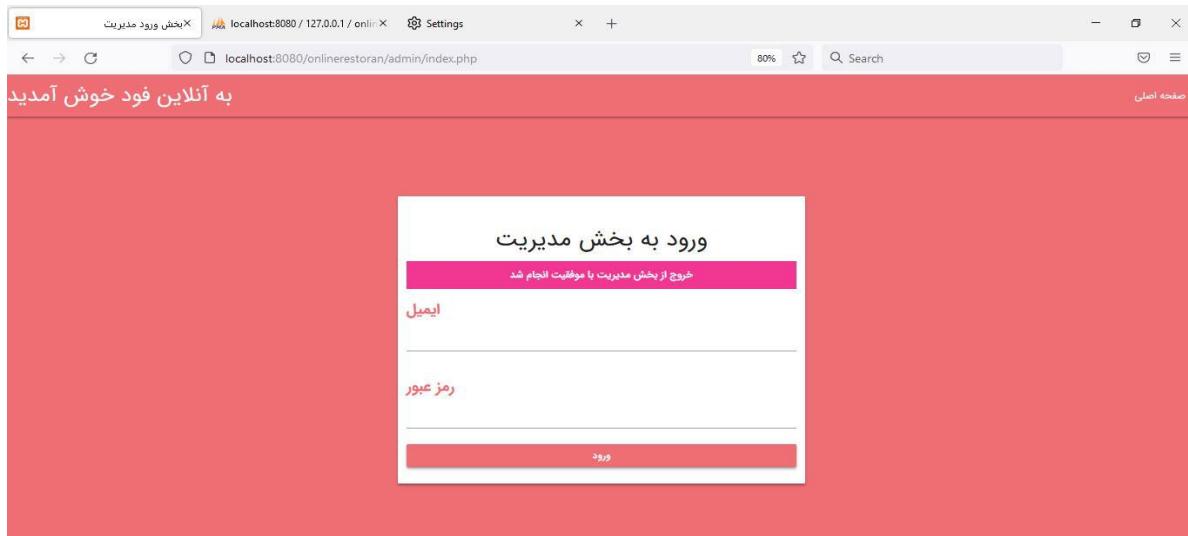


Figure 10-2-4: Admin Login Page

References

Reference to the website

<https://www.w3schools.com/>

<https://www.php.net/>

<https://www.geeksforgeeks.org/php/>

<https://www.tutorialspoint.com/php/index.htm>

<https://www.phpclasses.org/>

<https://www.codecademy.com/learn/learn-php>

<https://chishi.ir/863-kabab-torki/>

<https://snappfood.ir/>

Refer to the textbook

The book of software engineering by Roger S. Pressman, 7th edition