

Weather API Key (staff_nav.php:311)

```
307     });
308
309     //Weather API
310     function fetchWeatherData(latitude, longitude) {
311 +       const apiKey = 'YOUR_API_KEY';
312       const apiUrl = 'https://api.openweathermap.org/data/2.5/weather?lat=${latitude}&lon=${longitude}&units=metric&appid=${apiKey}';
313
314       console.log("Fetching weather data from URL:", apiUrl);
315
316       fetch(apiUrl)
317         .then(response => {
318           console.log("API response status:", response.status);
319           return response.json();
```

Google Maps API Key (add_restaurant.php:193)

```
189     <img class="modal-content" id="img01">
190     <div id="caption"></div>
191   </div>
192
193   <script src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&libraries=places"></script>
194   <script>
195     function initAutocomplete() {
196       const addressField = document.getElementById('address');
197       const autocomplete = new google.maps.places.Autocomplete(addressField);
198
199       autocomplete.addListener('place_changed', function() {
200         const place = autocomplete.getPlace();
201         if (!place.geometry) {
202           return;
203         }
```

(active_account.php:189)

```
183     <br />
184     <a href="link_account.php">Link To Other Company</a>
185   </form>
186 </div>
187 </div>
188
189 <script src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&libraries=places"></script>
190 <script>
191   function initAutocomplete() {
192     const addressField = document.getElementById('address');
193     const autocomplete = new google.maps.places.Autocomplete(addressField);
194
195     autocomplete.addListener('place_changed', function() {
196       const place = autocomplete.getPlace();
197       if (!place.geometry) {
198         return;
199       }
200     }
```

(seller_edit_information.php:245)

```
239 +   <div id="myModal" class="modal">
240     <span class="close">&times;</span>
241     <img class="modal-content" id="img01">
242     <div id="caption"></div>
243   </div>
244
245   <script src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&libraries=places"></script>
246   <script>
247     // Function to delete images from the documents
248     function deleteImage(button, imageUrl) {
249       var imageWrapper = button.closest('.image-wrapper');
250       imageWrapper.remove();
251
252       var deletedImagesInput = document.getElementById('deleted_images');
253       var deletedImages = deletedImagesInput.value ? deletedImagesInput.value.split(',') : [];
254
255       // Add the imageUrl to the list of deleted images
256       deletedImages.push(imageUrl);
```

(add_address.php:60)

```
55 +      </form>
56      </div>
57
58      <?php include '../partials/footer.php'; ?>
59  </body>
60  <script src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&libraries=places"></script>
61  <script>
62      function initAutocomplete() {
63          const addressField = document.getElementById('line2');
64          const autocomplete = new google.maps.places.Autocomplete(addressField);
65
66          autocomplete.setFields(['address_component', 'geometry']);
67
68          autocomplete.addListener('place_changed', function() {
69              const place = autocomplete.getPlace();
```

(edit_address.php:110)

```
106
107      <?php include '../partials/footer.php'; ?>
108  </body>
109
110  <script src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&libraries=places"></script>
111  <script>
112      function initAutocomplete() {
113          const addressField = document.getElementById('line2');
114          const autocomplete = new google.maps.places.Autocomplete(addressField);
115
116          autocomplete.setFields(['address_component', 'geometry']);
117
```

PHP Mailer (mailer.php:18,19)

```
14
15      $mail->Host = "smtp.gmail.com";
16      $mail->SMTPSecure = PHPMailer::ENCRYPTION_SMTPS;
17      $mail->Port = 465;
18      $mail->Username = "YOUR_EMAIL_ADDRESS";
19      $mail->Password = "YOUR_PASSWORD";
20
21      $mail->isHTML(true);
22
23      return $mail;
24
```

(send_password_reset.php:31)

```
27 +
28  if ($stmt->rowCount() > 0) {
29      $mail = require __DIR__ . "/mailer.php";
30
31      $mail->setFrom("email_address");
32      $mail->addAddress($email);
33      $mail->Subject = "Password Reset";
34      $mail->Body = <<<END
35      <p>Click <a href="http://localhost:3000/login_management/reset_password.php?token=$token">here</a> to reset your password.</p>
36      END;
37
```

Database

resource > Database.php > ...

```
1  <?php
2
3  $username = 'root';
4  $dsn = 'mysql:host=localhost; dbname=makan_apa';
5  $password = '';
6
7  try {
8      $db = new PDO($dsn, $username, $password);
9
10     $db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
11
12     $db->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_ASSOC);
13
14 } catch (PDOException $ex) {
15     echo "Connected failed ".$ex->getMessage();
16 }
17
```

Paypal (admin_wallet.php:30)

```
21 + <!DOCTYPE html>
22 <html lang="en">
23
24 <head>
25     <meta charset="utf-8">
26     <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
27     <link href="../../vendor/fontawesome-free/css/all.min.css" rel="stylesheet" type="text/css">
28     <link href="https://fonts.googleapis.com/css?family=Nunito:200,200i,300,300i,400,400i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">
29     <link href="../../vendor/datatables/dataTables.bootstrap4.min.css" rel="stylesheet">
30     <script src="https://www.paypal.com/sdk/js?client-id=YOUR_API_KEY&currency=MYR"></script>
31     <style>
32         .container-fluid {
33             margin-bottom: 5%;
34         }
35     </style>
36
```

(payment.php:62)

```
58 </div>
59 <?php include '../partials/footer.php'; ?>
60
61 <!-- Load the PayPal script -->
62 <script src="https://www.paypal.com/sdk/js?client-id=YOUR_API_KEY&currency=MYR"></script>
63 <script>
64     paypal.Buttons({
65         createOrder: function(data, actions) {
66             return actions.order.create({
67                 purchase_units: [{
68                     amount: {
69                         value: 100
70                     }
71                 }]
72             });
73         },
74         onApprove: function(data, actions) {
75             return actions.order.capture();
76         }
77     });
78
```