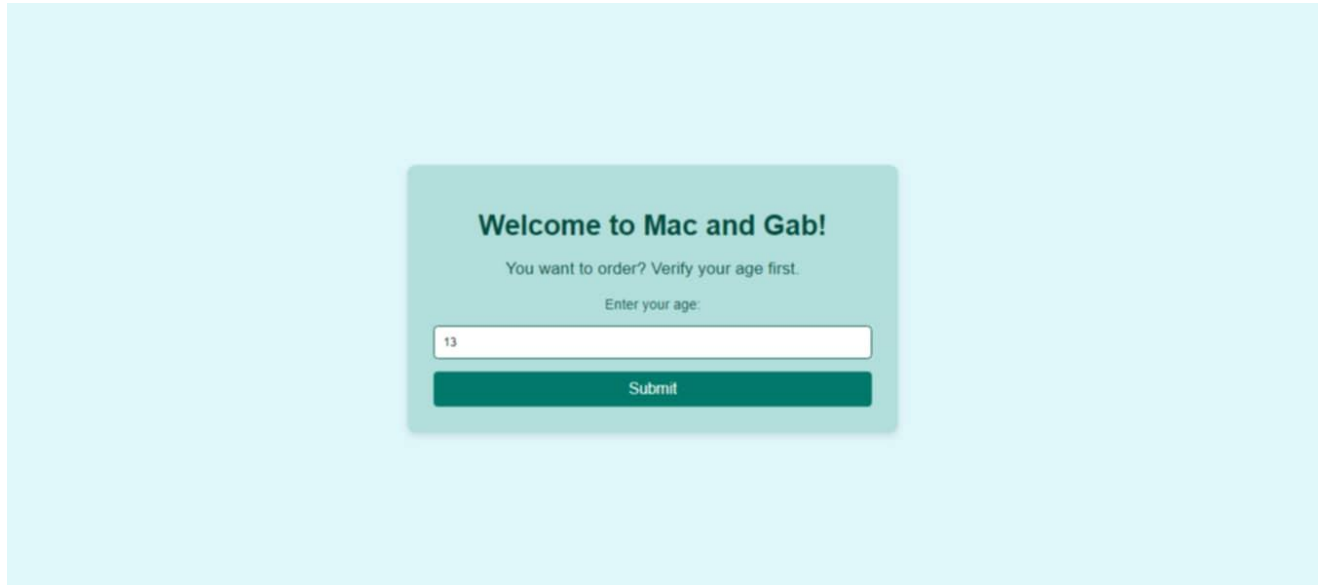


KIM ANTONETTE BRON

BSIT 3C

WEB DEVELOPMENT

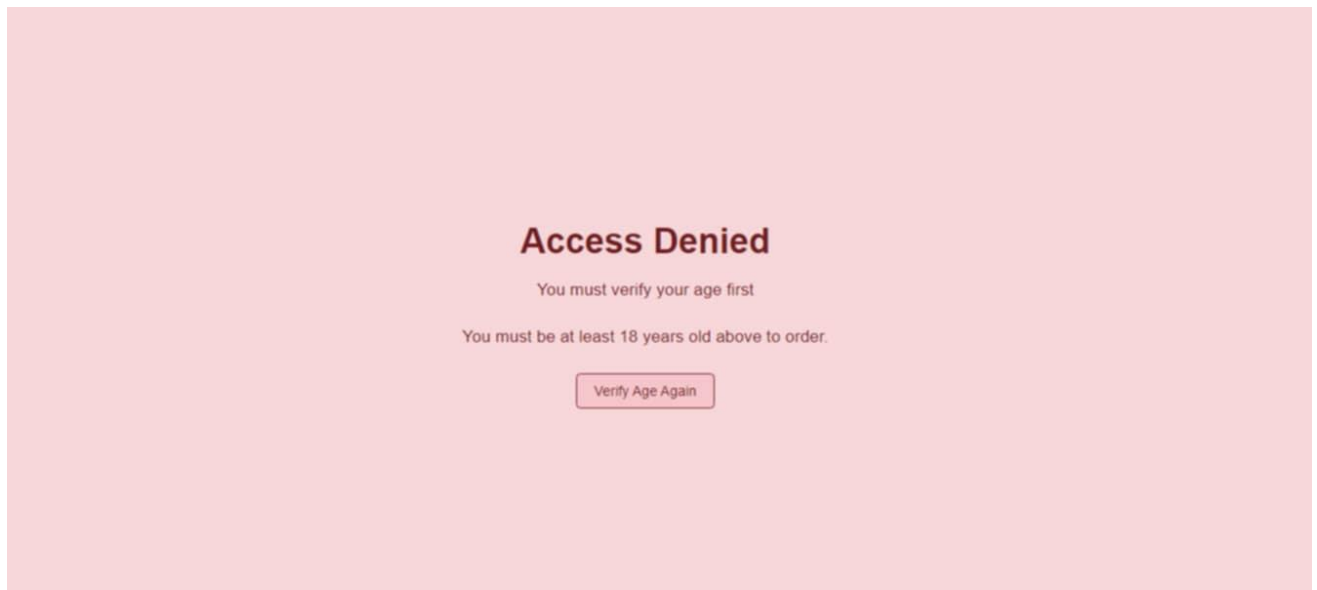
RENDERED WEBPAGES

A screenshot of a web page with a light blue background. In the center, there is a teal-colored rounded rectangle. Inside this rectangle, the text "Welcome to Mac and Gab!" is displayed in a bold, dark teal font. Below this, in a smaller, lighter teal font, is the text "You want to order? Verify your age first." Further down, the text "Enter your age:" is shown. Below this text is a white input field with a thin border, containing the number "13". At the bottom of the teal rectangle is a dark teal button with the word "Submit" in white text.

This is the initial form where the user is prompted to verify their age, before allowing them to proceed.

As shown in the input field, the user is asked to enter their age.

Below the input field is a submit button for the user to submit their entered age.

A screenshot of a web page with a light pink background. In the center, the text "Access Denied" is displayed in a bold, dark red font. Below this, in a smaller, lighter pink font, is the text "You must verify your age first". Further down, the text "You must be at least 18 years old above to order." is shown. At the bottom center is a light pink button with rounded corners and the text "Verify Age Again" in a darker pink font.

This will appear after the user has submitted their age and it has been determined that they do not meet the minimum age requirement (likely 18 years or older) and access to the ordering system is denied.

The user can try again by clicking the button and reentering a valid age.

KERNEL

```
1  <?php
2
3  namespace App\Http;
4
5  use Illuminate\Foundation\Http\Kernel as HttpKernel;
6
7  class Kernel extends HttpKernel
8  {
9      protected $routeMiddleware = [
10         // ...
11         'CheckAge' => \App\Http\Middleware\CheckAge::class,
12     ];
13     protected $middlewareGroups = [
14         'web' => [
15             // ...
16             \App\Http\Middleware\LogRequests::class,
17         ],
18         // ...
19     ];
20
21
22 }
```

- **namespace App\Http;** - defines the namespace for the Kernel class, indicating that it belongs to the App\Http namespace.
- **class Kernel extends HttpKernel** – responsible for handling HTTP requests and middleware.
- **protected \$routeMiddleware** - an associative array where you register route-specific middleware that can be applied to individual routes.
- **'CheckAge' => \App\Http\Middleware\CheckAge::class** - registers the CheckAge middleware. The key 'CheckAge' is a name you can use to reference this middleware when defining routes. The value \App\Http\Middleware\CheckAge::class points to the actual middleware class.
- **protected \$middlewareGroups** - This property allows you to group middleware together so that you can apply multiple middleware to routes with a single designation.
- **'web' => [...]** - defines a middleware group called web. Middleware within this group will be applied to routes that are defined within the web context.
- **\App\Http\Middleware\LogRequests::class** - This adds the LogRequests middleware to the web group, meaning that any route that uses this middleware group will automatically have request logging enabled.

Log Requests

```
1  <?php
2
3  namespace App\Http\Middleware;
4
5  use Closure;
6  use Illuminate\Http\Request;
7  use Illuminate\Support\Facades\Log;
8
9  class LogRequests
10 {
11     /**
12      * Handle an incoming request.
13      *
14      * @param \Illuminate\Http\Request $request
15      * @param \Closure $next
16      * @return mixed
17      */
18     public function handle(Request $request, Closure $next)
19     {
20         $logData = '[' . now() . ']' . ' ' . $request->method() . ' ' . $request->fullUrl();
21         Log::channel('custom')->info($logData);
22         return $next($request);
23     }
24 }
```

- **namespace App\Http\Middleware;** - specifies that the **LogRequests** class belongs to the App\Http\Middleware namespace, which organizes the code into logical groups.

Use Statements:

- use **Closure**:: Imports the Closure class, which is a special type of PHP class for anonymous functions.
- use **Illuminate\Http\Request**; - Imports the Request class, which provides an interface for handling HTTP requests.
- use **Illuminate\Support\Facades\Log**; -Imports the Log facade, which provides logging capabilities.
- class **LogRequests** - declares the LogRequests class, which will handle the logging of incoming HTTP requests.
- **public function handle(Request \$request, Closure \$next)** - is the entry point for the middleware and is automatically called when a request is received.
- **Parameters:**
 - **\$request**: An instance of the Request class representing the incoming HTTP request.
 - **\$next**: A closure that represents the next middleware in the stack or the final destination of the request.
- **Log::channel('custom')->info(\$logData);** - This line logs the constructed log message to a custom log channel defined in your logging configuration. The info method is used to log informational messages.
- **return \$next(\$request);** -This line calls the next middleware in the stack, allowing the request to continue its journey through the application. If this is the last middleware, it will pass the request to the intended route/controller.

CheckAge

```
1  <?php
2  |
3  namespace App\Http\Middleware;
4
5  use Closure;
6  use Illuminate\Http\Request;
7
8  class CheckAge
9  {
10     /**
11      * Handle an incoming request.
12      *
13      * @param \Illuminate\Http\Request $request
14      * @param \Closure $next
15      * @param int $minAge
16      * @return mixed
17      */
18
19     public function handle(Request $request, Closure $next, $minAge = 18)
20     {
21         $age = $request->query('age');
22
23         // If the age is less than the minimum age, redirect to access denied
24         if ($age < $minAge) {
25             return redirect('/access-denied');
26         }
27
28         // If the age is 18 or older, let the request continue to home page
29         if ($age < 21) {
30             return redirect('/');
31         }
32         // If age is 21 or older continue to home page
33     }
```

The **middleware** belongs to the App\Http\Middleware namespace.

It **imports** Closure and Illuminate\Http\Request from the Laravel framework.

The class **CheckAge** is defined to handle incoming HTTP requests and check the user's age.

The **handle()** method where the main logic resides, takes three parameters:

- Request **\$request**: The incoming request object that **contains user data**.
- Closure **\$next**: A **closure** that represents the next step in the request lifecycle.
- int **\$minAge**: A **default parameter** that **defines the minimum age requirement** (default is 18).

Age Validation:

- The age is retrieved from the query string using **\$request->query('age')**.
- **Condition 1: If age is less than the minimum age (\$minAge):**
 - A middleware checks if the age is below the defined \$minAge (18). If it is, the user is redirected to an access-denied page (/access-denied).
- **Condition 2: If age is 18 or older but below 21:**
 - If the age is less than 21 but equal to or greater than the minimum age (18), the user is redirected to the home page (/).

- **Condition 3: If age is 21 or older:**
 - If the age is 21 or older, the request continues to the next step in the lifecycle (\$next(\$request)), allowing further processing.

ROUTES

```
// Route for checking age (form submission)
Route::get('/check-age', function (Request $request) {
    $age = $request->query('age'); // Get age from query parameter

    // Redirect to the welcome route with the age
    return redirect()->route('welcome', ['age' => $age]);
})->name('check.age');

// The welcome route with CheckAge middleware applied
Route::middleware(['check.age']->get('/welcome', function (Request $request) {
    return view('welcome', ['age' => $request->query('age')]); // Pass the age to the view
})->name('welcome');

// Access Denied route
Route::get('/access-denied', function () {
    return view('access-denied'); // Corrected view name
})->name('access.denied');

});
```

Route::get('/', function () {

- This route listens for a **GET** request at the /check-age URL.
- It **retrieves the age** from the query string using \$request->query('age').
- After getting the age, it **redirects** the user to the welcome route and passes the age as a parameter.

Route::middleware(['check.age']->get('/welcome', function (Request \$request) {

- This route listens for a **GET** request at the /welcome URL.
- It applies the **CheckAge** middleware
- The middleware ensures that the user meets the age requirement before proceeding to this route.
- The age is passed to the welcome view to be displayed on the page.

Route::get('/access-denied', function () {

- This route listens for a **GET** request at the /access-denied URL.
- It renders the **access-denied** view, which would show a message informing the user that they are not allowed to proceed.