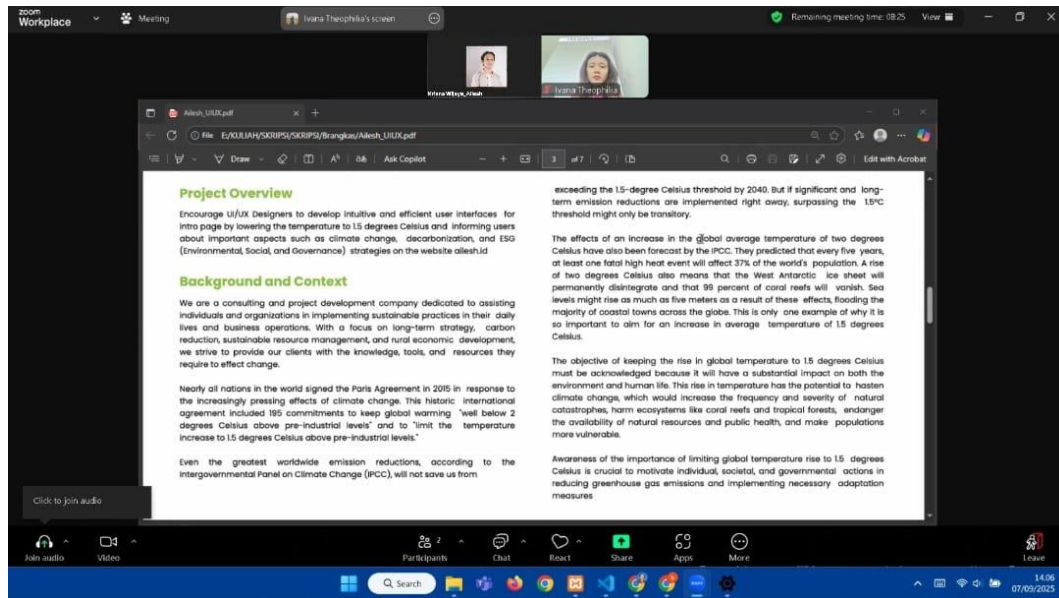
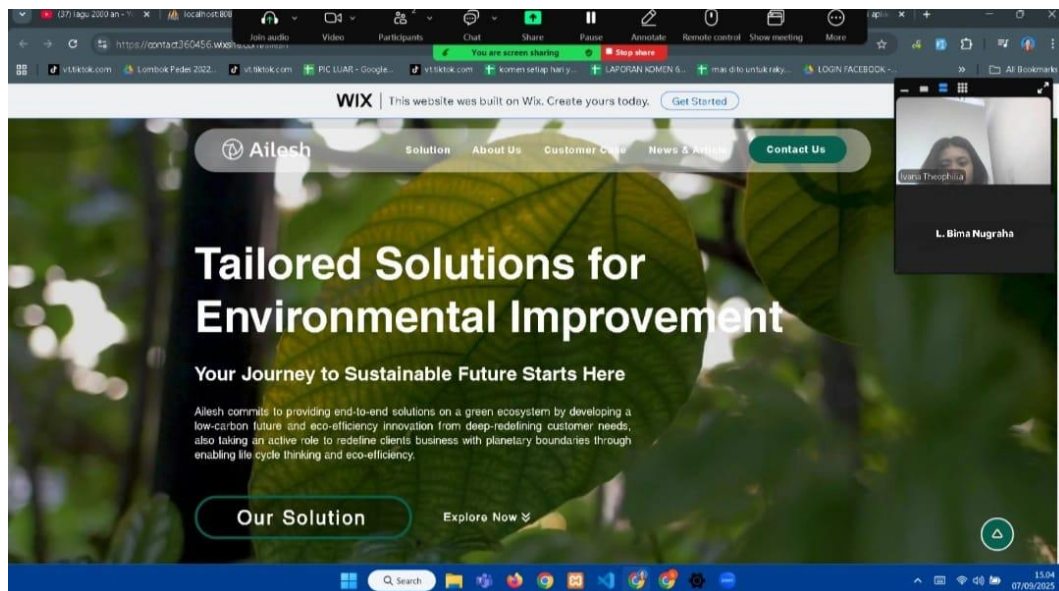


LAMPIRAN

Lampiran 1. Wawancara dengan Krisna Wijaya (*creative design Ailesh*)

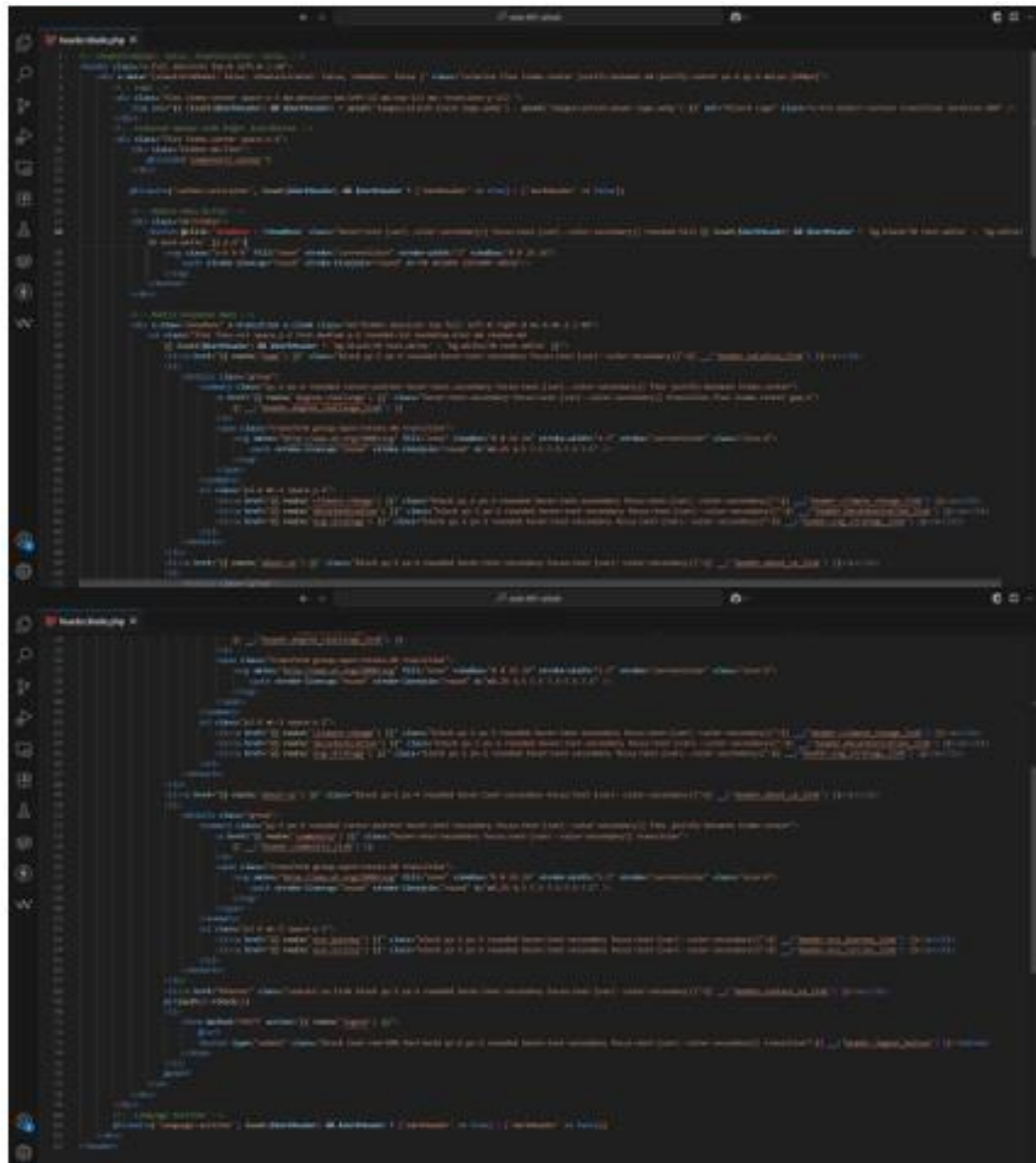


Lampiran 2. Wawancara dengan L.Bima Nugraha (*user persona*)



Lampiran 3. *Coding header*

Header, komponen ini bertanggung jawab untuk menampilkan bagian *header*, berisi logo, menu navigasi, tombol kalkulator karbon, dan pilihan bahasa. Tampilannya *responsive* untuk *desktop* dan *mobile*.



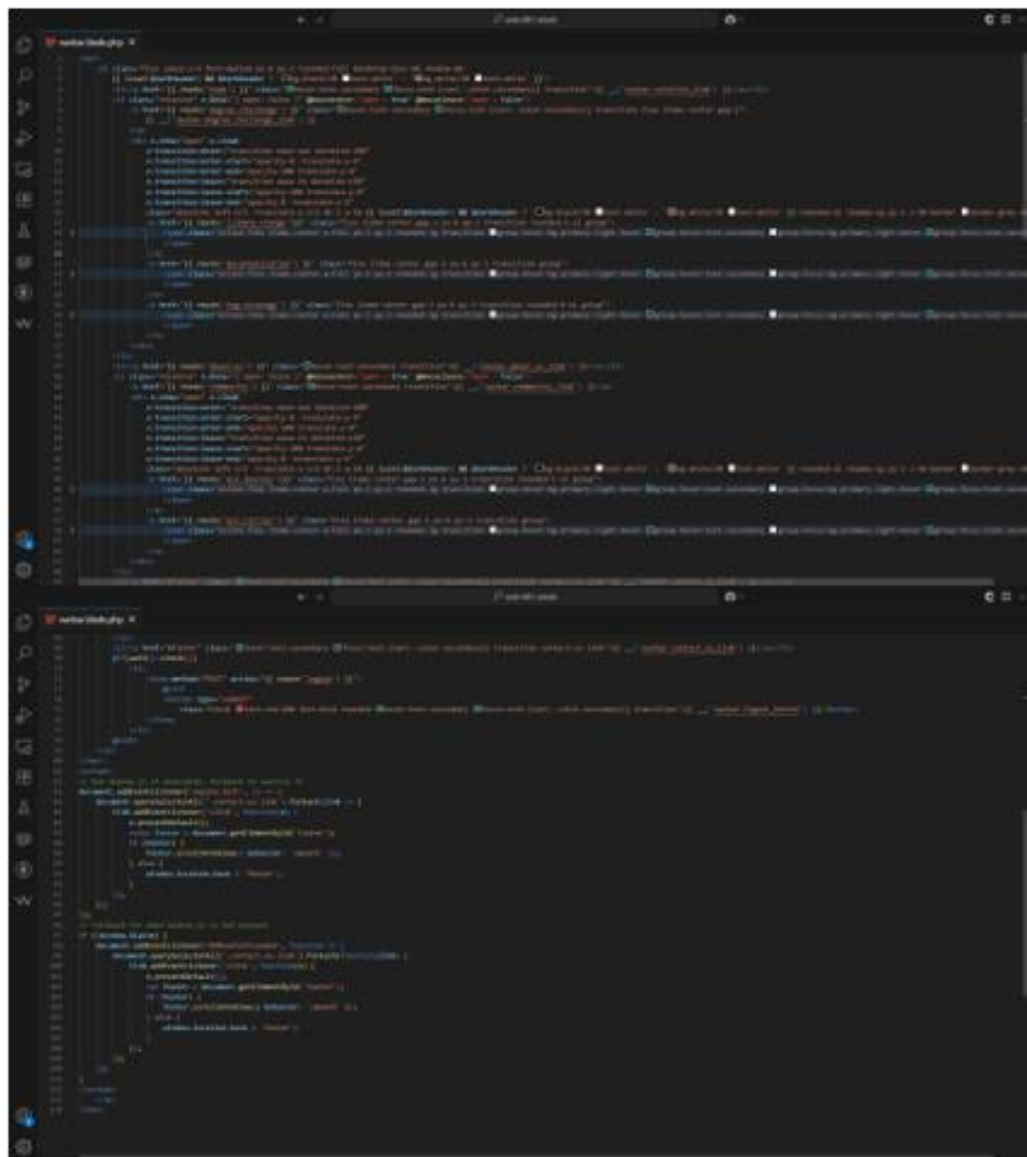
Lampiran 4. Coding footerr

Footer, komponen ini adalah bagian kaki (*footer*) situs *web*, di dalamnya terdapat informasi kontak seperti alamat *email* dan nomor telepon, tautan ke media sosial, serta menu navigasi tambahan. Berfungsi sebagai penutup halaman yang memberikan informasi penting dan tautan cepat kepada pengunjung.

```
1 <footer id="footer" class="bg-secondary text-white py-12">
2   <div class="px-4 lg:px:100px">
3     <div class="flex flex-col md:flex-row justify-between gap-8">
4       <!-- Left Column: Logo and Address -->
5       <div class="flex flex-col gap-6 max-w-lg md:max-w-md">
6         <div>
7           
8         </div>
9         <div class="flex flex-col md:flex-row gap-2 justify-start">
10          <div class="flex flex-col gap-2 w-18">
11            <p class="text-md font-medium leading-tight">
12              {{ __('Footer Address') }}
13            </p>
14          </div>
15          <div class="flex flex-col justify-start">
16            <div class="text-lg font-sonbold mb-2">{{ __('Footer speak to us') }}
```

Lampiran 5. Coding navbar

Navbar, komponen yang berisi navigasi utama untuk situs web, yang dipanggil di dalam *header.blade.php* untuk tampilan *desktop*. Terdapat *link* ke halaman-halaman utama dan beberapa memiliki *submenu dropdown*.

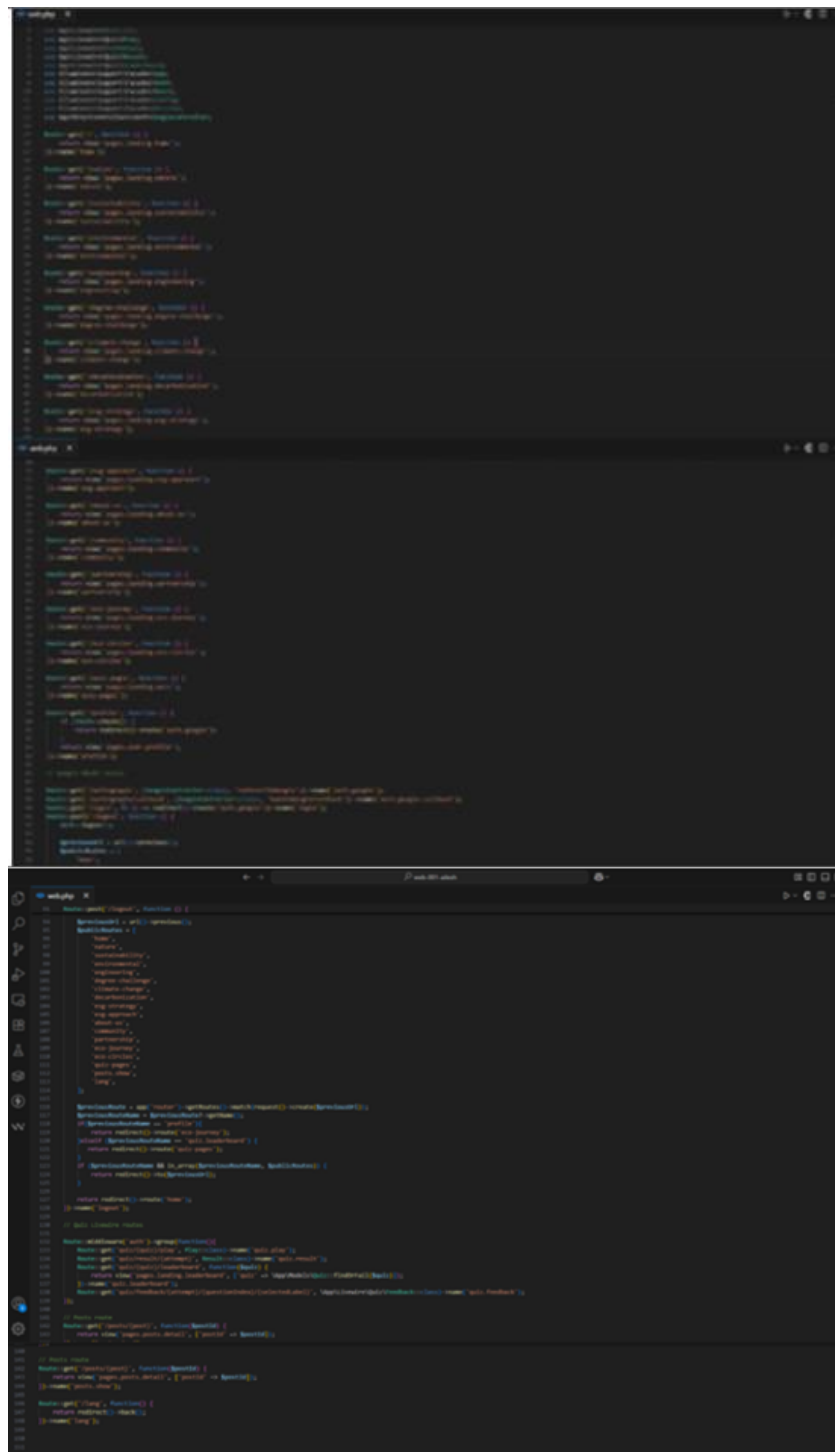


```
// navbar.blade.php
<div class="container">
    <div class="row">
        <div class="col-12">
            <div class="navbar">
                <div class="navbar-brand">
                    <a href="#">Logo</a>
                </div>
                <div class="navbar-collapse">
                    <ul class="list-unstyled">
                        <li><a href="#">Home</a></li>
                        <li><a href="#">About</a></li>
                        <li><a href="#">Services</a></li>
                        <li><a href="#">Contact</a></li>
                        <li><a href="#">Dropdown</a>
                            <ul class="list-unstyled">
                                <li><a href="#">Dropdown Item 1</a></li>
                                <li><a href="#">Dropdown Item 2</a></li>
                                <li><a href="#">Dropdown Item 3</a></li>
                                <li><a href="#">Dropdown Item 4</a></li>
                                <li><a href="#">Dropdown Item 5</a></li>
                            </ul>
                        </li>
                    </ul>
                </div>
            </div>
        </div>
    </div>
</div>
```

```
// navbar.blade.php
<div class="container">
    <div class="row">
        <div class="col-12">
            <div class="navbar">
                <div class="navbar-brand">
                    <a href="#">Logo</a>
                </div>
                <div class="navbar-collapse">
                    <ul class="list-unstyled">
                        <li><a href="#">Home</a></li>
                        <li><a href="#">About</a></li>
                        <li><a href="#">Services</a></li>
                        <li><a href="#">Contact</a></li>
                        <li><a href="#">Dropdown</a>
                            <ul class="list-unstyled">
                                <li><a href="#">Dropdown Item 1</a></li>
                                <li><a href="#">Dropdown Item 2</a></li>
                                <li><a href="#">Dropdown Item 3</a></li>
                                <li><a href="#">Dropdown Item 4</a></li>
                                <li><a href="#">Dropdown Item 5</a></li>
                            </ul>
                        </li>
                    </ul>
                </div>
            </div>
        </div>
    </div>
</div>
```

Lampiran 6. Coding route

Route, tempat mengatur dan mendefinisikan semua URL yang akan diakses *user* melalui *browser*.



```
import express from 'express';
import { body, query, param } from 'express-validator';
import { validationResult } from 'express-validator';
import { authMiddleware } from './middleware/auth';
import { userMiddleware } from './middleware/user';
import { validate } from './utils/validate';
import { generateToken } from './utils/generateToken';
import { login } from './controllers/auth';
import { register } from './controllers/auth';
import { getUserProfile } from './controllers/user';
import { updateUserProfile } from './controllers/user';
import { deleteUserProfile } from './controllers/user';
import { getAllUsers } from './controllers/user';
import { getUserById } from './controllers/user';
import { updateUserById } from './controllers/user';
import { deleteUserById } from './controllers/user';
import { createPost } from './controllers/post';
import { getAllPosts } from './controllers/post';
import { getPostById } from './controllers/post';
import { updatePostById } from './controllers/post';
import { deletePostById } from './controllers/post';
import { createComment } from './controllers/comment';
import { getAllComments } from './controllers/comment';
import { getCommentById } from './controllers/comment';
import { updateCommentById } from './controllers/comment';
import { deleteCommentById } from './controllers/comment';
import { createLike } from './controllers/like';
import { getAllLikes } from './controllers/like';
import { getLikeById } from './controllers/like';
import { updateLikeById } from './controllers/like';
import { deleteLikeById } from './controllers/like';
import { createFollow } from './controllers/follow';
import { getAllFollows } from './controllers/follow';
import { getFollowById } from './controllers/follow';
import { updateFollowById } from './controllers/follow';
import { deleteFollowById } from './controllers/follow';
import { createBlock } from './controllers/block';
import { getAllBlocks } from './controllers/block';
import { getBlockById } from './controllers/block';
import { updateBlockById } from './controllers/block';
import { deleteBlockById } from './controllers/block';
import { createReport } from './controllers/report';
import { getAllReports } from './controllers/report';
import { getReportById } from './controllers/report';
import { updateReportById } from './controllers/report';
import { deleteReportById } from './controllers/report';
import { createFeedback } from './controllers/feedback';
import { getAllFeedbacks } from './controllers/feedback';
import { getFeedbackById } from './controllers/feedback';
import { updateFeedbackById } from './controllers/feedback';
import { deleteFeedbackById } from './controllers/feedback';

const router = express.Router();

// Authentication routes
router.post('/login', login);
router.post('/register', register);

// User management routes
router.get('/profile', authMiddleware, getUserProfile);
router.put('/profile', authMiddleware, updateUserProfile);
router.delete('/profile', authMiddleware, deleteUserProfile);
router.get('/users', getAllUsers);
router.get('/user/:id', getUserById);
router.put('/user/:id', updateUserById);
router.delete('/user/:id', deleteUserById);

// Post management routes
router.post('/posts', createPost);
router.get('/posts', getAllPosts);
router.get('/post/:id', getPostById);
router.put('/post/:id', updatePostById);
router.delete('/post/:id', deletePostById);

// Comment management routes
router.post('/comments', createComment);
router.get('/comments', getAllComments);
router.get('/comment/:id', getCommentById);
router.put('/comment/:id', updateCommentById);
router.delete('/comment/:id', deleteCommentById);

// Like management routes
router.post('/likes', createLike);
router.get('/likes', getAllLikes);
router.get('/like/:id', getLikeById);
router.put('/like/:id', updateLikeById);
router.delete('/like/:id', deleteLikeById);

// Follow management routes
router.post('/follows', createFollow);
router.get('/follows', getAllFollows);
router.get('/follow/:id', getFollowById);
router.put('/follow/:id', updateFollowById);
router.delete('/follow/:id', deleteFollowById);

// Block management routes
router.post('/blocks', createBlock);
router.get('/blocks', getAllBlocks);
router.get('/block/:id', getBlockById);
router.put('/block/:id', updateBlockById);
router.delete('/block/:id', deleteBlockById);

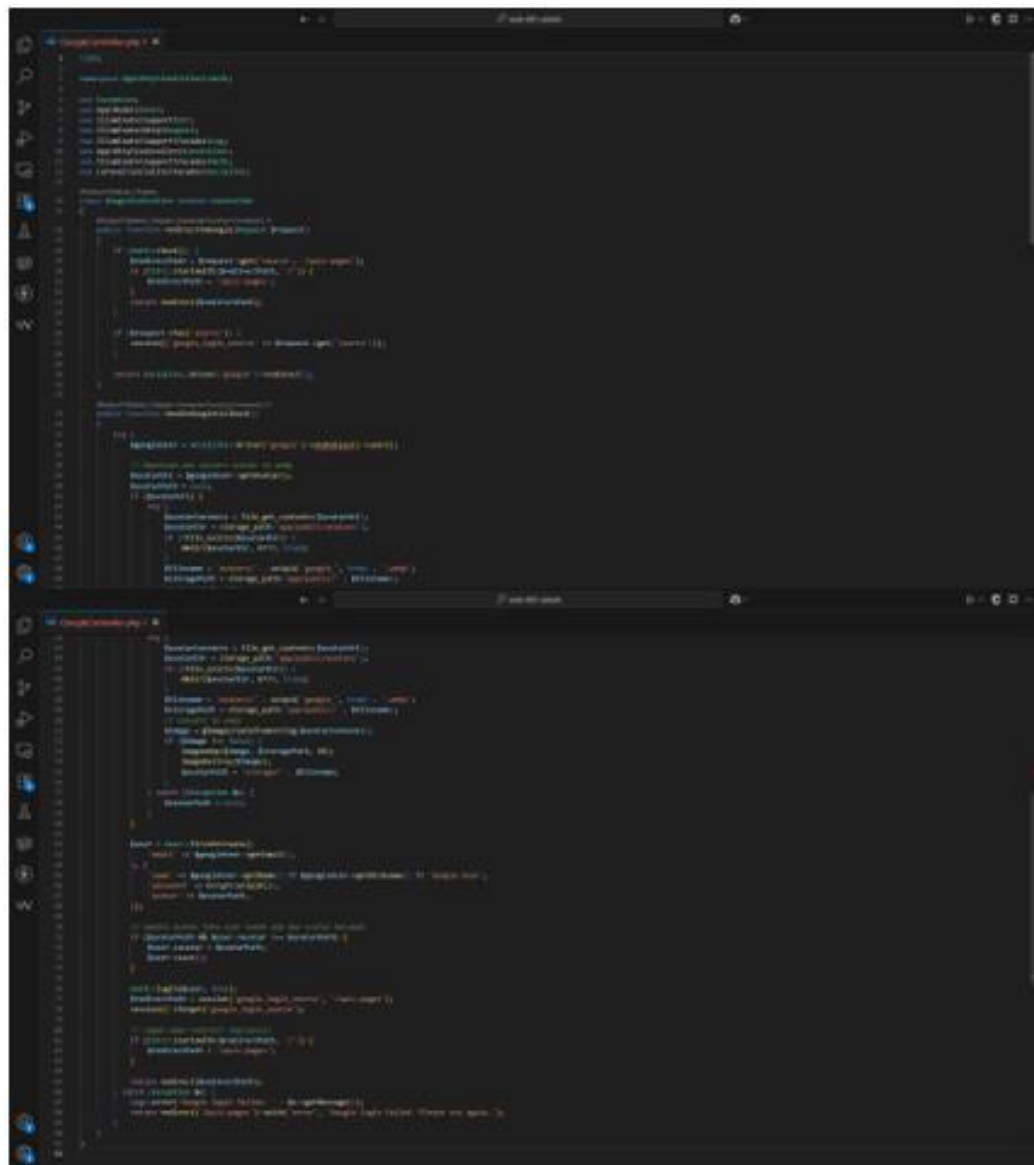
// Report management routes
router.post('/reports', createReport);
router.get('/reports', getAllReports);
router.get('/report/:id', getReportById);
router.put('/report/:id', updateReportById);
router.delete('/report/:id', deleteReportById);

// Feedback management routes
router.post('/feedbacks', createFeedback);
router.get('/feedbacks', getAllFeedbacks);
router.get('/feedback/:id', getFeedbackById);
router.put('/feedback/:id', updateFeedbackById);
router.delete('/feedback/:id', deleteFeedbackById);

export default router;
```

Lampiran 7. Coding google aunt

Google aunt, komponen ini untuk mengatur autentikasi *user* menggunakan teknologi *oAuth google*.



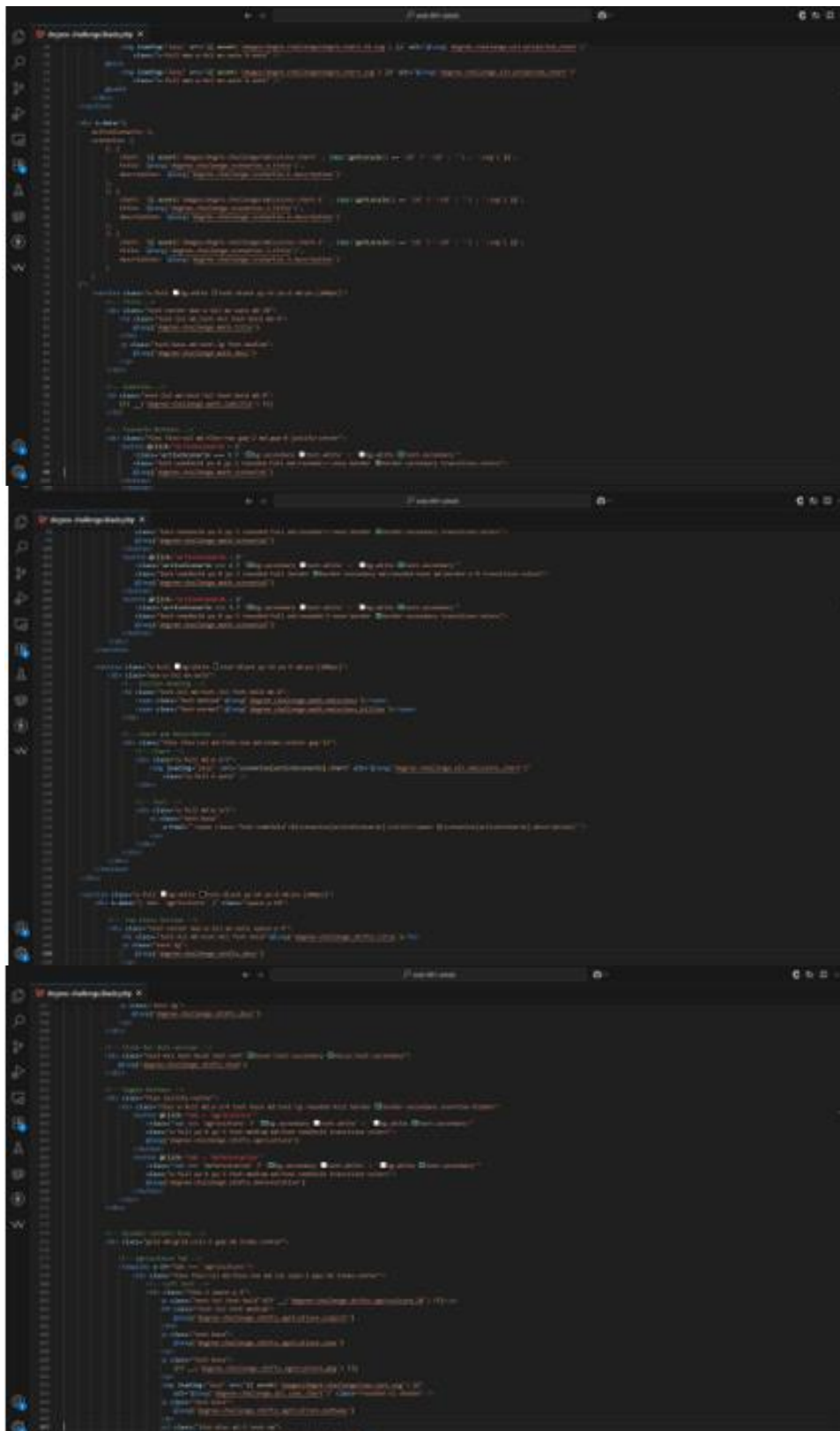
Lampiran 8. Coding layout base

Layout base, komponen ini berisi kerangka (*layout*) utama dari semua halaman. Semua halaman lain akan dimasukkan ke dalam kerangka ini, memastikan tampilan yang konsisten.

```
1 <docType html>
2 <html lang="en" class="scroll-smooth">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title{{ config('app.name', 'Alish') }}</title>
7   <link href="https://fonts.googleapis.com/css?family=Montserrat" rel="stylesheet">
8   <style>
9     @keyframes spin-slow {
10       from { transform: rotate(0deg); }
11       to { transform: rotate(360deg); }
12     }
13     .animate-spin-slow {
14       animation: spin-slow 18s linear infinite;
15     }
16   </style>
17   @vite(['resources/css/app.css', 'resources/js/app.js'])
18   @livewireStyles
19 </head>
20 <body class="{{ $bg-white ? 'bg-white' : 'text-black' }}">
21   <main class="">
22     @yield('content')
23   </main>
24   @include('components.tips')
25   @livewireScripts
26   <script src="https://cdn.jsdelivr.net/npm/alpinejs" defer=</script>
27   @stack('scripts')
28 </body>
29 </html>
```

Lampiran 9. Halaman 1,5 Degree Challenge

```
1 @extends('layouts.app')
2
3 @section('content')
4   <section class="relative h-screen bg-secondary flex flex-col px-6 py-18 md:py-16 md:px-[100px]">
5     @include('components.header')
6     <div class="flex flex-1 flex-col md:flex-row items-center justify-start md:justify-between mt-5 md:mt-8">
7       <!-- Left: Text Content -->
8       <div class="md:w-1/2 w-full md:mt-12 md:mt-8">
9         <h1 class="text-4xl md:text-3xl font-bold text-white mb-6" @lang('degree-challenge.hero.title')>Hi!
10         <p class="text-white text-base font-medium md:text-lg mb-4">
11           @lang('degree-challenge.hero.desc')
12       </div>
13       <!-- Right: Image Content -->
14       <div class="md:w-1/2 w-full flex justify-center items-center">
15         
17       </div>
18     </div>
19   </section>
20
21   <section class="md:min-h-screen bg-white px-6 md:px-[100px] py-28">
22     <!-- Top Headings -->
23     <div class="text-center max-w-3xl mx-auto mb-16">
24       <h2 class="text-3xl md:text-2xl font-bold text-black mb-4">
25         @lang('degree-challenge.how.title')
26       </h2>
27       <p class="text-base text-center font-medium md:text-lg text-neutral-800 leading-relaxed">
28         @lang('degree-challenge.how.desc')
29       </p>
30     </div>
31
32     <!-- Section Title -->
33     <h3 class="text-2xl md:text-2xl font-bold text-black mb-6">
34       @lang('degree-challenge.how.decade')
35     </h3>
36
37     <div class="mb-8">
38       <p class="text-base font-bold md:text-lg text-gray-600 leading-relaxed">
39         @lang('degree-challenge.how.projected')
40       </p>
41       <p class="text-base font-medium md:text-lg text-gray-500 leading-relaxed">
42         @lang('degree-challenge.how.billion')
43       </p>
44     </div>
45
46     <!-- Chart Placeholder -->
47     <div class="w-full overflow-auto">
48       @if ($app->getOption('new-1d'))
49         
50       @else
51         
52       @endif
53     </div>
```




```

1  // Import the necessary modules
2  import { Router } from 'express';
3  import { authMiddleware } from './middleware/auth';
4  import { loginController } from './controllers/login';
5  import { registerController } from './controllers/register';
6  import { forgotPasswordController } from './controllers/forgotPassword';
7  import { resetPasswordController } from './controllers/resetPassword';
8  import { verifyEmailController } from './controllers/verifyEmail';
9  import { refreshTokenController } from './controllers/refreshToken';
10 import { logoutController } from './controllers/logout';
11
12 // Create a new Express router
13 const router = Router();
14
15 // Apply the authMiddleware to all routes
16 router.use(authMiddleware);
17
18 // Define the routes
19 router.post('/login', loginController);
20 router.post('/register', registerController);
21 router.post('/forgot-password', forgotPasswordController);
22 router.post('/reset-password', resetPasswordController);
23 router.post('/verify-email', verifyEmailController);
24 router.post('/refresh-token', refreshTokenController);
25 router.post('/logout', logoutController);
26
27 // Export the router
28 export default router;

```

```

1  // Import the necessary modules
2  import { Router } from 'express';
3  import { authMiddleware } from './middleware/auth';
4  import { loginController } from './controllers/login';
5  import { registerController } from './controllers/register';
6  import { forgotPasswordController } from './controllers/forgotPassword';
7  import { resetPasswordController } from './controllers/resetPassword';
8  import { verifyEmailController } from './controllers/verifyEmail';
9  import { refreshTokenController } from './controllers/refreshToken';
10 import { logoutController } from './controllers/logout';
11
12 // Create a new Express router
13 const router = Router();
14
15 // Apply the authMiddleware to all routes
16 router.use(authMiddleware);
17
18 // Define the routes
19 router.post('/login', loginController);
20 router.post('/register', registerController);
21 router.post('/forgot-password', forgotPasswordController);
22 router.post('/reset-password', resetPasswordController);
23 router.post('/verify-email', verifyEmailController);
24 router.post('/refresh-token', refreshTokenController);
25 router.post('/logout', logoutController);
26
27 // Export the router
28 export default router;

```

```

1  // Import the necessary modules
2  import { Router } from 'express';
3  import { authMiddleware } from './middleware/auth';
4  import { loginController } from './controllers/login';
5  import { registerController } from './controllers/register';
6  import { forgotPasswordController } from './controllers/forgotPassword';
7  import { resetPasswordController } from './controllers/resetPassword';
8  import { verifyEmailController } from './controllers/verifyEmail';
9  import { refreshTokenController } from './controllers/refreshToken';
10 import { logoutController } from './controllers/logout';
11
12 // Create a new Express router
13 const router = Router();
14
15 // Apply the authMiddleware to all routes
16 router.use(authMiddleware);
17
18 // Define the routes
19 router.post('/login', loginController);
20 router.post('/register', registerController);
21 router.post('/forgot-password', forgotPasswordController);
22 router.post('/reset-password', resetPasswordController);
23 router.post('/verify-email', verifyEmailController);
24 router.post('/refresh-token', refreshTokenController);
25 router.post('/logout', logoutController);
26
27 // Export the router
28 export default router;

```



```

1  // ...
2  // ...
3  // ...
4  // ...
5  // ...
6  // ...
7  // ...
8  // ...
9  // ...
10 // ...
11 // ...
12 // ...
13 // ...
14 // ...
15 // ...
16 // ...
17 // ...
18 // ...
19 // ...
20 // ...
21 // ...
22 // ...
23 // ...
24 // ...
25 // ...
26 // ...
27 // ...
28 // ...
29 // ...
30 // ...
31 // ...
32 // ...
33 // ...
34 // ...
35 // ...
36 // ...
37 // ...
38 // ...
39 // ...
40 // ...
41 // ...
42 // ...
43 // ...
44 // ...
45 // ...
46 // ...
47 // ...
48 // ...
49 // ...
50 // ...
51 // ...
52 // ...
53 // ...
54 // ...
55 // ...
56 // ...
57 // ...
58 // ...
59 // ...
60 // ...
61 // ...
62 // ...
63 // ...
64 // ...
65 // ...
66 // ...
67 // ...
68 // ...
69 // ...
70 // ...
71 // ...
72 // ...
73 // ...
74 // ...
75 // ...
76 // ...
77 // ...
78 // ...
79 // ...
80 // ...
81 // ...
82 // ...
83 // ...
84 // ...
85 // ...
86 // ...
87 // ...
88 // ...
89 // ...
90 // ...
91 // ...
92 // ...
93 // ...
94 // ...
95 // ...
96 // ...
97 // ...
98 // ...
99 // ...
100 // ...

```

Lampiran 10. Halaman *Climate Change*

```

1  // ...
2  // ...
3  // ...
4  // ...
5  // ...
6  // ...
7  // ...
8  // ...
9  // ...
10 // ...
11 // ...
12 // ...
13 // ...
14 // ...
15 // ...
16 // ...
17 // ...
18 // ...
19 // ...
20 // ...
21 // ...
22 // ...
23 // ...
24 // ...
25 // ...
26 // ...
27 // ...
28 // ...
29 // ...
30 // ...
31 // ...
32 // ...
33 // ...
34 // ...
35 // ...
36 // ...
37 // ...
38 // ...
39 // ...
40 // ...
41 // ...
42 // ...
43 // ...
44 // ...
45 // ...
46 // ...
47 // ...
48 // ...
49 // ...
50 // ...
51 // ...
52 // ...
53 // ...
54 // ...
55 // ...
56 // ...
57 // ...
58 // ...
59 // ...
60 // ...
61 // ...
62 // ...
63 // ...
64 // ...
65 // ...
66 // ...
67 // ...
68 // ...
69 // ...
70 // ...
71 // ...
72 // ...
73 // ...
74 // ...
75 // ...
76 // ...
77 // ...
78 // ...
79 // ...
80 // ...
81 // ...
82 // ...
83 // ...
84 // ...
85 // ...
86 // ...
87 // ...
88 // ...
89 // ...
90 // ...
91 // ...
92 // ...
93 // ...
94 // ...
95 // ...
96 // ...
97 // ...
98 // ...
99 // ...
100 // ...

```

[illegible]

```

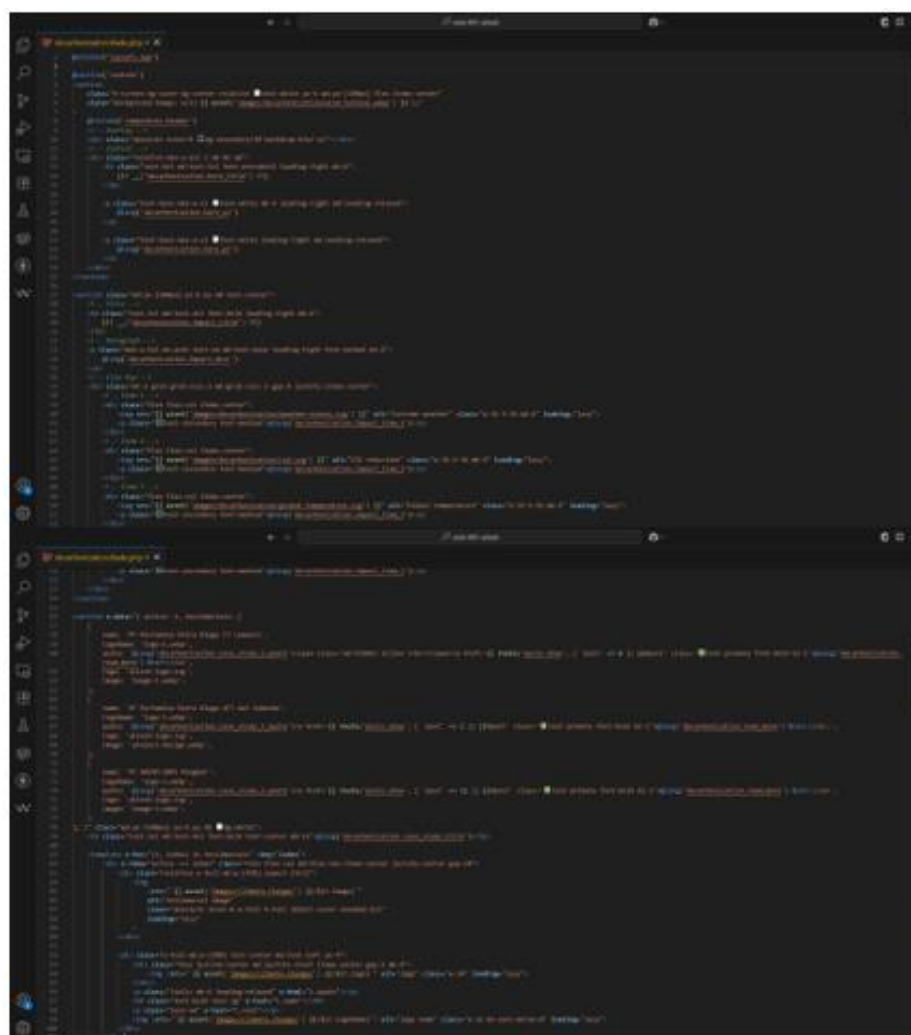
1  #!/usr/bin/perl -w
2
3  use strict;
4  use warnings;
5
6  my $url = "https://www.example.com";
7  my $user_agent = "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.114 Safari/537.36";
8
9  # Create a new HTTP client
10 my $client = LWP::UserAgent->new(
11     agent => $user_agent,
12     ssl_opts => {
13         verify => 1,
14     },
15 );
16
17 # Fetch the HTML content
18 my $response = $client->get($url);
19
20 # Check if the request was successful
21 if ($response->is_success) {
22     # Extract the HTML content
23     my $html = $response->content;
24
25     # Parse the HTML content using HTML::TreeBuilder
26     my $tree = HTML::TreeBuilder->new_from_content($html);
27
28     # Traverse the tree and extract the desired data
29     my $data = extract_data($tree);
30
31     # Print the extracted data
32     print "Extracted data: " . $data . "\n";
33 } else {
34     # Handle the error
35     my $error_message = "Failed to fetch the page: " . $response->status_line;
36     print $error_message . "\n";
37 }
38
39 # Function to extract data from the HTML tree
40 sub extract_data {
41     my $tree = shift;
42
43     # Extract the title
44     my $title = $tree->find_one('h1');
45     my $title_text = $title->as_text() if $title;
46
47     # Extract the main content
48     my $main_content = $tree->find_one('div');
49     my $main_content_text = $main_content->as_text() if $main_content;
50
51     # Return the extracted data
52     return $title_text . " " . $main_content_text;
53 }
54
55 # Run the script
56 __END__
57
58 #!/usr/bin/perl -w
59
60 use strict;
61 use warnings;
62
63 my $url = "https://www.example.com";
64 my $user_agent = "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.114 Safari/537.36";
65
66 # Create a new HTTP client
67 my $client = LWP::UserAgent->new(
68     agent => $user_agent,
69     ssl_opts => {
70         verify => 1,
71     },
72 );
73
74 # Fetch the HTML content
75 my $response = $client->get($url);
76
77 # Check if the request was successful
78 if ($response->is_success) {
79     # Extract the HTML content
80     my $html = $response->content;
81
82     # Parse the HTML content using HTML::TreeBuilder
83     my $tree = HTML::TreeBuilder->new_from_content($html);
84
85     # Traverse the tree and extract the desired data
86     my $data = extract_data($tree);
87
88     # Print the extracted data
89     print "Extracted data: " . $data . "\n";
90 } else {
91     # Handle the error
92     my $error_message = "Failed to fetch the page: " . $response->status_line;
93     print $error_message . "\n";
94 }
95
96 # Function to extract data from the HTML tree
97 sub extract_data {
98     my $tree = shift;
99
100    # Extract the title
101    my $title = $tree->find_one('h1');
102    my $title_text = $title->as_text() if $title;
103
104    # Extract the main content
105    my $main_content = $tree->find_one('div');
106    my $main_content_text = $main_content->as_text() if $main_content;
107
108    # Return the extracted data
109    return $title_text . " " . $main_content_text;
110 }
111
112 # Run the script
113 __END__

```

[illegible]



Lampiran 11. Halaman *Decarbonization*



```

1  #!/usr/bin/perl
2
3  use strict;
4  use warnings;
5
6  my $url = "http://www.example.com";
7  my $user_agent = "Mozilla/5.0 (Windows NT 6.0; rv:2.0.1) Gecko/20100101 Firefox/4.0.1";
8
9  my $response = LWP::Simple::get($url);
10
11  if ($response) {
12      my $content = $response;
13
14      # Extract the title
15      my $title = $content =~ /<title>(.*?)</title>/i ? $1 : "Unknown Title";
16
17      # Extract the meta description
18      my $description = $content =~ /<meta name="description" content="(.*?)"/i ? $1 : "No meta description found";
19
20      # Extract the meta keywords
21      my $keywords = $content =~ /<meta name="keywords" content="(.*?)"/i ? $1 : "No meta keywords found";
22
23      # Print the results
24      print "Title: $title\n";
25      print "Description: $description\n";
26      print "Keywords: $keywords\n";
27  } else {
28      print "Failed to retrieve the page content.\n";
29  }
30
31  exit 0;
32
33  # Example usage:
34  # perl http_title_desc_key.pl http://www.example.com
35
36  # Output:
37  # Title: Example Title
38  # Description: Example Description
39  # Keywords: Example Keywords
40
41  # Error handling:
42  # perl http_title_desc_key.pl http://www.example.com 2>&1
43  # perl http_title_desc_key.pl http://www.example.com 2>&1 | grep -i error
44
45  # License: MIT
46
47  # Author: John Doe
48  # Contact: john.doe@example.com
49
50  # Version: 1.0
51
52  # Copyright (c) 2010 John Doe. All rights reserved.
53
54  # This program is free software; you can redistribute it and/or
55  # modify it under the terms of the GNU General Public License
56  # as published by the Free Software Foundation; either version 2
57  # of the License, or (at your option) any later version.
58
59  # This program is distributed in the hope that it will be useful,
60  # but WITHOUT ANY WARRANTY; without even the implied warranty of
61  # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
62  # GNU General Public License for more details.
63
64  # You should have received a copy of the GNU General Public License
65  # along with this program; if not, write to the Free Software
66  # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
67  # MA 02110-1301, USA.
68
69  # End of file
70
71  # Example usage:
72  # perl http_title_desc_key.pl http://www.example.com
73
74  # Output:
75  # Title: Example Title
76  # Description: Example Description
77  # Keywords: Example Keywords
78
79  # Error handling:
80  # perl http_title_desc_key.pl http://www.example.com 2>&1
81  # perl http_title_desc_key.pl http://www.example.com 2>&1 | grep -i error
82
83  # License: MIT
84
85  # Author: John Doe
86  # Contact: john.doe@example.com
87
88  # Version: 1.0
89
90  # Copyright (c) 2010 John Doe. All rights reserved.
91
92  # This program is free software; you can redistribute it and/or
93  # modify it under the terms of the GNU General Public License
94  # as published by the Free Software Foundation; either version 2
95  # of the License, or (at your option) any later version.
96
97  # This program is distributed in the hope that it will be useful,
98  # but WITHOUT ANY WARRANTY; without even the implied warranty of
99  # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
100 # GNU General Public License for more details.
101
102 # You should have received a copy of the GNU General Public License
103 # along with this program; if not, write to the Free Software
104 # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
105 # MA 02110-1301, USA.
106
107 # End of file
108
109 # Example usage:
110 # perl http_title_desc_key.pl http://www.example.com
111
112 # Output:
113 # Title: Example Title
114 # Description: Example Description
115 # Keywords: Example Keywords
116
117 # Error handling:
118 # perl http_title_desc_key.pl http://www.example.com 2>&1
119 # perl http_title_desc_key.pl http://www.example.com 2>&1 | grep -i error
120
121 # License: MIT
122
123 # Author: John Doe
124 # Contact: john.doe@example.com
125
126 # Version: 1.0
127
128 # Copyright (c) 2010 John Doe. All rights reserved.
129
130 # This program is free software; you can redistribute it and/or
131 # modify it under the terms of the GNU General Public License
132 # as published by the Free Software Foundation; either version 2
133 # of the License, or (at your option) any later version.
134
135 # This program is distributed in the hope that it will be useful,
136 # but WITHOUT ANY WARRANTY; without even the implied warranty of
137 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
138 # GNU General Public License for more details.
139
140 # You should have received a copy of the GNU General Public License
141 # along with this program; if not, write to the Free Software
142 # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
143 # MA 02110-1301, USA.
144
145 # End of file
146
147 # Example usage:
148 # perl http_title_desc_key.pl http://www.example.com
149
150 # Output:
151 # Title: Example Title
152 # Description: Example Description
153 # Keywords: Example Keywords
154
155 # Error handling:
156 # perl http_title_desc_key.pl http://www.example.com 2>&1
157 # perl http_title_desc_key.pl http://www.example.com 2>&1 | grep -i error
158
159 # License: MIT
160
161 # Author: John Doe
162 # Contact: john.doe@example.com
163
164 # Version: 1.0
165
166 # Copyright (c) 2010 John Doe. All rights reserved.
167
168 # This program is free software; you can redistribute it and/or
169 # modify it under the terms of the GNU General Public License
170 # as published by the Free Software Foundation; either version 2
171 # of the License, or (at your option) any later version.
172
173 # This program is distributed in the hope that it will be useful,
174 # but WITHOUT ANY WARRANTY; without even the implied warranty of
175 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
176 # GNU General Public License for more details.
177
178 # You should have received a copy of the GNU General Public License
179 # along with this program; if not, write to the Free Software
180 # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
181 # MA 02110-1301, USA.
182
183 # End of file
184
185 # Example usage:
186 # perl http_title_desc_key.pl http://www.example.com
187
188 # Output:
189 # Title: Example Title
190 # Description: Example Description
191 # Keywords: Example Keywords
192
193 # Error handling:
194 # perl http_title_desc_key.pl http://www.example.com 2>&1
195 # perl http_title_desc_key.pl http://www.example.com 2>&1 | grep -i error
196
197 # License: MIT
198
199 # Author: John Doe
200 # Contact: john.doe@example.com
201
202 # Version: 1.0
203
204 # Copyright (c) 2010 John Doe. All rights reserved.
205
206 # This program is free software; you can redistribute it and/or
207 # modify it under the terms of the GNU General Public License
208 # as published by the Free Software Foundation; either version 2
209 # of the License, or (at your option) any later version.
210
211 # This program is distributed in the hope that it will be useful,
212 # but WITHOUT ANY WARRANTY; without even the implied warranty of
213 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
214 # GNU General Public License for more details.
215
216 # You should have received a copy of the GNU General Public License
217 # along with this program; if not, write to the Free Software
218 # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
219 # MA 02110-1301, USA.
220
221 # End of file
222
223 # Example usage:
224 # perl http_title_desc_key.pl http://www.example.com
225
226 # Output:
227 # Title: Example Title
228 # Description: Example Description
229 # Keywords: Example Keywords
230
231 # Error handling:
232 # perl http_title_desc_key.pl http://www.example.com 2>&1
233 # perl http_title_desc_key.pl http://www.example.com 2>&1 | grep -i error
234
235 # License: MIT
236
237 # Author: John Doe
238 # Contact: john.doe@example.com
239
240 # Version: 1.0
241
242 # Copyright (c) 2010 John Doe. All rights reserved.
243
244 # This program is free software; you can redistribute it and/or
245 # modify it under the terms of the GNU General Public License
246 # as published by the Free Software Foundation; either version 2
247 # of the License, or (at your option) any later version.
248
249 # This program is distributed in the hope that it will be useful,
250 # but WITHOUT ANY WARRANTY; without even the implied warranty of
251 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
252 # GNU General Public License for more details.
253
254 # You should have received a copy of the GNU General Public License
255 # along with this program; if not, write to the Free Software
256 # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
257 # MA 02110-1301, USA.
258
259 # End of file
260
261 # Example usage:
262 # perl http_title_desc_key.pl http://www.example.com
263
264 # Output:
265 # Title: Example Title
266 # Description: Example Description
267 # Keywords: Example Keywords
268
269 # Error handling:
270 # perl http_title_desc_key.pl http://www.example.com 2>&1
271 # perl http_title_desc_key.pl http://www.example.com 2>&1 | grep -i error
272
273 # License: MIT
274
275 # Author: John Doe
276 # Contact: john.doe@example.com
277
278 # Version: 1.0
279
280 # Copyright (c) 2010 John Doe. All rights reserved.
281
282 # This program is free software; you can redistribute it and/or
283 # modify it under the terms of the GNU General Public License
284 # as published by the Free Software Foundation; either version 2
285 # of the License, or (at your option) any later version.
286
287 # This program is distributed in the hope that it will be useful,
288 # but WITHOUT ANY WARRANTY; without even the implied warranty of
289 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
290 # GNU General Public License for more details.
291
292 # You should have received a copy of the GNU General Public License
293 # along with this program; if not, write to the Free Software
294 # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
295 # MA 02110-1301, USA.
296
297 # End of file
298
299 # Example usage:
300 # perl http_title_desc_key.pl http://www.example.com
301
302 # Output:
303 # Title: Example Title
304 # Description: Example Description
305 # Keywords: Example Keywords
306
307 # Error handling:
308 # perl http_title_desc_key.pl http://www.example.com 2>&1
309 # perl http_title_desc_key.pl http://www.example.com 2>&1 | grep -i error
310
311 # License: MIT
312
313 # Author: John Doe
314 # Contact: john.doe@example.com
315
316 # Version: 1.0
317
318 # Copyright (c) 2010 John Doe. All rights reserved.
319
320 # This program is free software; you can redistribute it and/or
321 # modify it under the terms of the GNU General Public License
322 # as published by the Free Software Foundation; either version 2
323 # of the License, or (at your option) any later version.
324
325 # This program is distributed in the hope that it will be useful,
326 # but WITHOUT ANY WARRANTY; without even the implied warranty of
327 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
328 # GNU General Public License for more details.
329
330 # You should have received a copy of the GNU General Public License
331 # along with this program; if not, write to the Free Software
332 # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
333 # MA 02110-1301, USA.
334
335 # End of file
336
337 # Example usage:
338 # perl http_title_desc_key.pl http://www.example.com
339
340 # Output:
341 # Title: Example Title
342 # Description: Example Description
343 # Keywords: Example Keywords
344
345 # Error handling:
346 # perl http_title_desc_key.pl http://www.example.com 2>&1
347 # perl http_title_desc_key.pl http://www.example.com 2>&1 | grep -i error
348
349 # License: MIT
350
351 # Author: John Doe
352 # Contact: john.doe@example.com
353
354 # Version: 1.0
355
356 # Copyright (c) 2010 John Doe. All rights reserved.
357
358 # This program is free software; you can redistribute it and/or
359 # modify it under the terms of the GNU General Public License
360 # as published by the Free Software Foundation; either version 2
361 # of the License, or (at your option) any later version.
362
363 # This program is distributed in the hope that it will be useful,
364 # but WITHOUT ANY WARRANTY; without even the implied warranty of
365 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE
```

```

1  using System;
2  using System.IO;
3  class Program
4  {
5      static void Main()
6      {
7          string path = "C:\\Users\\user\\Documents\\input.txt";
8          string content = File.ReadAllText(path);
9          Console.WriteLine(content);
10     }
11 }

```

```

// Account creation
function createAccount() {
    // Create a new account
    const account = {
        name: document.getElementById('name').value,
        email: document.getElementById('email').value,
        password: document.getElementById('password').value,
        confirmPassword: document.getElementById('confirmPassword').value,
        role: 'user'
    };

    // Check if the password and confirmPassword are the same
    if (account.password !== account.confirmPassword) {
        alert('Passwords do not match');
        return;
    }

    // Check if the email is already in use
    if (isEmailInUse(account.email)) {
        alert('Email is already in use');
        return;
    }

    // Save the account to the database
    saveAccount(account);
}

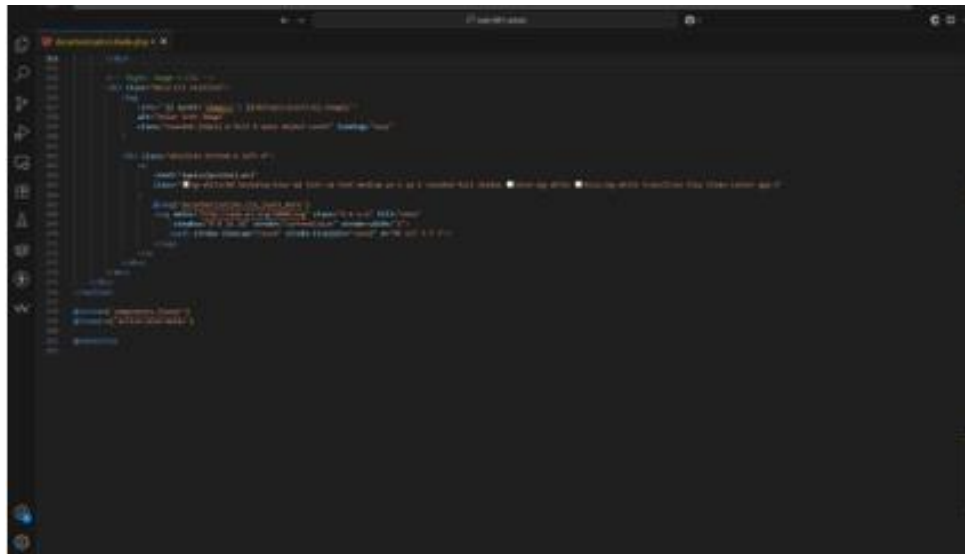
// Save account to the database
function saveAccount(account) {
    // Create a new document
    const doc = {
        name: account.name,
        email: account.email,
        password: account.password,
        confirmPassword: account.confirmPassword,
        role: account.role,
        createdAt: new Date().toISOString()
    };

    // Save the document to the database
    firebase.database().ref().push(doc);
}

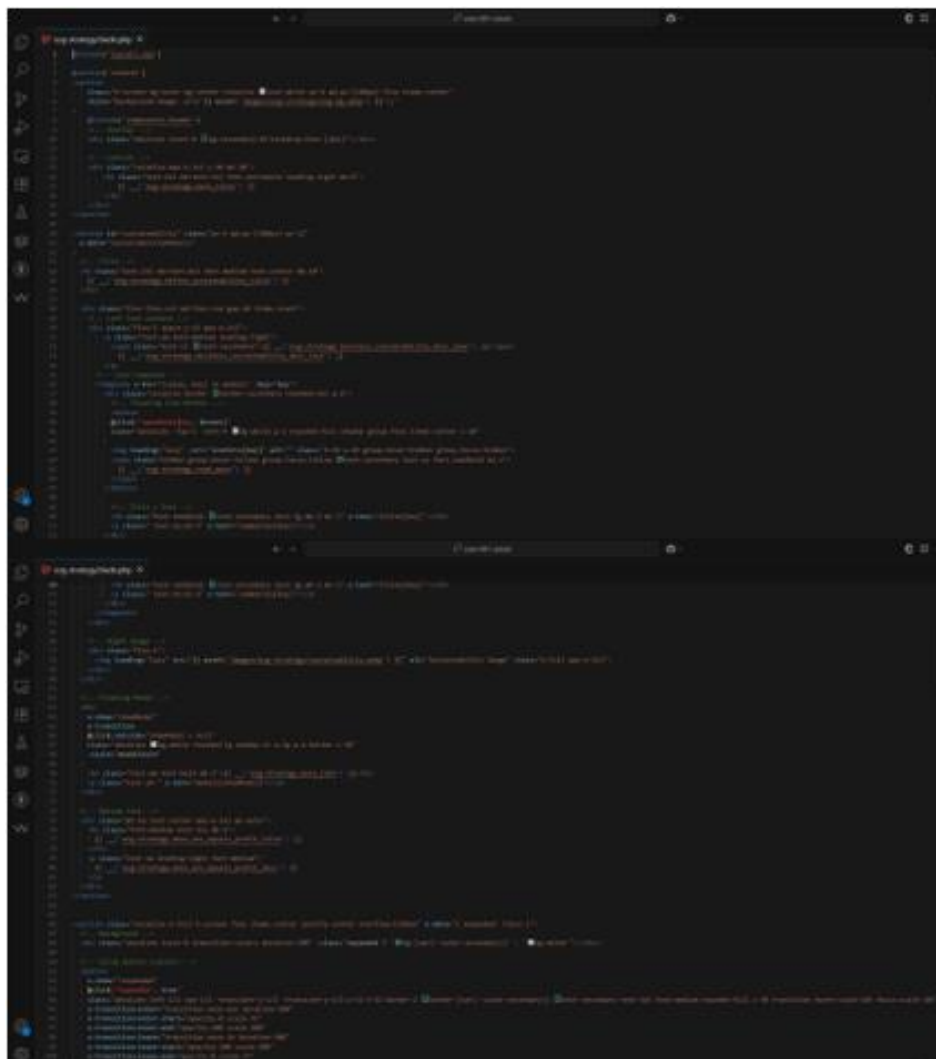
// Check if email is already in use
function isEmailInUse(email) {
    // Create a query
    const query = firebase.database().ref().orderBy('email').equalTo(email);

    // Get the results
    query.once('value', (snapshot) => {
        if (snapshot.exists()) {
            return true;
        }
        return false;
    });
}

```



Lampiran 12. Halaman *ESG Strategy*



```

1  #!/usr/bin/perl
2
3  use strict;
4  use warnings;
5
6  my $url = "http://www.example.com/";
7
8  # Create a new HTTP client
9  my $client = LWP::UserAgent->new(
10     agent => "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/109.0",
11     ssl_opts => {
12         SSL_VERIFY_PEER => 1,
13         SSL_VERIFY_FAIL_IF_NO_PEER_CERT => 1,
14     },
15 );
16
17 # Get the HTML content of the URL
18 my $response = $client->get($url);
19
20 # Check if the request was successful
21 if ($response->is_success) {
22     # Get the HTML content
23     my $html = $response->content;
24
25     # Parse the HTML content using HTML::TreeBuilder
26     my $tree = HTML::TreeBuilder->new_from_content($html);
27
28     # Find all links in the HTML
29     my @links = $tree->find_all('a');
30
31     # Print the links
32     foreach my $link (@links) {
33         my $href = $link->attr('href');
34         my $text = $link->as_text;
35         print "Link: $href, Text: $text\n";
36     }
37 }
38 else {
39     # Handle errors
40     my $status_code = $response->code;
41     my $reason_phrase = $response->reason_phrase;
42     print "Error: $status_code $reason_phrase\n";
43 }
44
45 # Close the client
46 $client->close;
47
48 # End of script
49

```

```

1  #!/usr/bin/perl
2
3  use strict;
4  use warnings;
5
6  my $url = "http://www.example.com/";
7
8  # Create a new HTTP client
9  my $client = LWP::UserAgent->new(
10     agent => "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/109.0",
11     ssl_opts => {
12         SSL_VERIFY_PEER => 1,
13         SSL_VERIFY_FAIL_IF_NO_PEER_CERT => 1,
14     },
15 );
16
17 # Get the HTML content of the URL
18 my $response = $client->get($url);
19
20 # Check if the request was successful
21 if ($response->is_success) {
22     # Get the HTML content
23     my $html = $response->content;
24
25     # Parse the HTML content using HTML::TreeBuilder
26     my $tree = HTML::TreeBuilder->new_from_content($html);
27
28     # Find all links in the HTML
29     my @links = $tree->find_all('a');
30
31     # Print the links
32     foreach my $link (@links) {
33         my $href = $link->attr('href');
34         my $text = $link->as_text;
35         print "Link: $href, Text: $text\n";
36     }
37 }
38 else {
39     # Handle errors
40     my $status_code = $response->code;
41     my $reason_phrase = $response->reason_phrase;
42     print "Error: $status_code $reason_phrase\n";
43 }
44
45 # Close the client
46 $client->close;
47
48 # End of script
49

```

```

1  #!/usr/bin/perl
2
3  use strict;
4  use warnings;
5
6  my $url = "http://www.example.com/";
7
8  # Create a new HTTP client
9  my $client = LWP::UserAgent->new(
10     agent => "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/109.0",
11     ssl_opts => {
12         SSL_VERIFY_PEER => 1,
13         SSL_VERIFY_FAIL_IF_NO_PEER_CERT => 1,
14     },
15 );
16
17 # Get the HTML content of the URL
18 my $response = $client->get($url);
19
20 # Check if the request was successful
21 if ($response->is_success) {
22     # Get the HTML content
23     my $html = $response->content;
24
25     # Parse the HTML content using HTML::TreeBuilder
26     my $tree = HTML::TreeBuilder->new_from_content($html);
27
28     # Find all links in the HTML
29     my @links = $tree->find_all('a');
30
31     # Print the links
32     foreach my $link (@links) {
33         my $href = $link->attr('href');
34         my $text = $link->as_text;
35         print "Link: $href, Text: $text\n";
36     }
37 }
38 else {
39     # Handle errors
40     my $status_code = $response->code;
41     my $reason_phrase = $response->reason_phrase;
42     print "Error: $status_code $reason_phrase\n";
43 }
44
45 # Close the client
46 $client->close;
47
48 # End of script
49

```



```
177 <a href="{{ route('esg-approach') }}" class="px-12 py-4 text-2xl font-sans border text-secondary border-secondary rounded-full transition-all duration-300">
178     {{ __("esg-strategy.get_started_button") }}
179 </a>
180 </div>
181 </div>
182 </section>
183 </div>
184 </script>
185
186 function sustainabilityModal() {
187     return {
188         showModal: null,
189         modalStyle: "",
190         models: {
191             env: '{{ __("esg-strategy.environmental_sustainability_summary") }}',
192             social: '{{ __("esg-strategy.social_sustainability_summary") }}',
193             econ: '{{ __("esg-strategy.economic_sustainability_summary") }}',
194         },
195         titles: {
196             env: '{{ __("esg-strategy.environmental_sustainability_title") }}',
197             social: '{{ __("esg-strategy.social_sustainability_title") }}',
198             econ: '{{ __("esg-strategy.economic_sustainability_title") }}',
199         },
200         summaries: {
201             env: '{{ __("esg-strategy.environmental_sustainability_summary") }}',
202             social: '{{ __("esg-strategy.social_sustainability_summary") }}',
203             econ: '{{ __("esg-strategy.economic_sustainability_summary") }}',
204         },
205         iconsSrcs: {
206             env: '{{ asset("images/esg-strategy/earth-icon.svg") }}',
207             social: '{{ asset("images/esg-strategy/community.svg") }}',
208             econ: '{{ asset("images/esg-strategy/economic.svg") }}',
209         },
210     };
211
212     openModal(key, event) {
213         this.showModal = key;
214
215         const rect = event.currentTarget.getBoundingClientRect();
216         this.modalStyle = "
217         top: ${rect.top + window.scrollY - 10}px;
218         left: ${rect.left + window.scrollX + rect.width + 12}px;
219     ";
220     };
221 </script>
222
223 @include('components.footer')
224
225 @endsection
```

Lampiran 13. Halaman *Community*

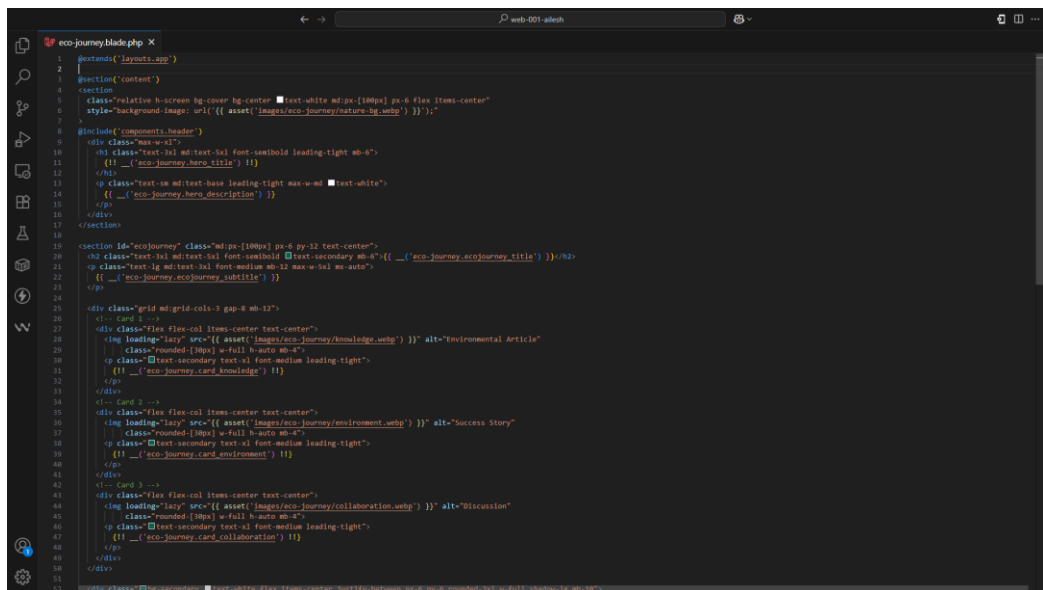
```
1 @extends('layouts.app')
2
3 @section('content')
4     <section class="relative h-screen w-full bg-cover bg-center flex items-center px-6 md:px-[100px] style="background-image: url('{{ asset("images/community/community-bg.webp") }}');">
5         @include('components.header')
6         <div class="absolute inset-0 bg-black/20"></div> <!-- Dark overlay -->
7
8         <div class="relative z-10 text-white max-w-lg">
9             <h1 class="text-2xl md:text-3xl font-bold mb-6">@lang('community.hero_title')</h1>
10             <div class="text-2xl md:text-3xl mb-4">
11                 @lang('community.hero_p1')
12             </div>
13             <div class="text-2xl md:text-3xl">
14                 @lang('community.hero_p2')
15             </div>
16         </div>
17     </section>
18
19     <section class="md:px-[100px] px-6 py-12 text-white">
20         <h2 class="text-2xl font-bold text-center mb-15">@lang('community.explore_title')</h2>
21
22         <div class="space-y-4">
23             <a href="{{ route('partnership') }}" class="relative bg-secondary rounded-dxl shadow-[0px_8px_20px_4px_primary] duration-300 hover:shadow-primary focus:shadow-primary h-[250px] flex items-center overflow-hidden">
24                 <div class="p-6 md:p-10 flex-shrink-0 h-16">
25                     <h3 class="text-2xl font-sans text-white">@lang('community.partnership')</h3>
26                 </div>
27                 
29             </a>
30             <a href="{{ route('eco-journey') }}" class="relative bg-secondary rounded-dxl shadow-[0px_8px_20px_4px_primary] duration-300 hover:shadow-secondary focus:shadow-secondary h-[250px] flex items-center overflow-hidden">
31                 
33                 <div class="p-6 md:p-10 flex-shrink-0 h-16 text-right">
34                     <h3 class="text-2xl font-sans text-white">@lang('community.eco_journey')</h3>
35                 </div>
36             </a>
37             <a href="{{ route('eco-circles') }}" class="relative bg-secondary rounded-dxl shadow-[0px_8px_20px_4px_primary] duration-300 hover:shadow-primary focus:shadow-primary h-[250px] flex items-center overflow-hidden">
38                 <div class="p-6 md:p-10 flex-shrink-0 h-16">
39                     <h3 class="text-2xl font-sans text-white">@lang('community.eco_circles')</h3>
40                 </div>
41                 
43             </a>
44         </div>
45     </section>
46
47 @include('components.footer')
48 @endsection
```

Lampiran 14. Halaman *Ecojourney*

Halaman ini terdapat *call to action* untuk berbagi cerita, diikuti dengan daftar postingan yang ditangani oleh komponen Livewire `PostList`. Berikut penjelasan detail fitur *post* :

Fitur *post* dibangun menggunakan beberapa komponen Livewire :

- 1) *PostList.php* : Menampilkan daftar postingan dari komunitas. Komponen ini menyediakan opsi filter berdasarkan "*all*", "*newest*", dan "*relevant*". Selain itu, terdapat tombol "*load more*" untuk memuat postingan tambahan.
- 2) *PostDetail.php* : Menampilkan detail dari sebuah postingan, mencakup judul, deskripsi, media, jumlah *likes*, serta komentar. Komponen ini juga memungkinkan pengguna menambahkan komentar pada postingan.
- 3) *Story.php* : Menyediakan formulir bagi pengguna untuk membuat postingan baru. Formulir ini mencakup isian judul, deskripsi, serta media berupa gambar. Saat *submit*, komponen ini akan membuat postingan baru dan menyimpannya ke *database*.



```
1 @extends('layouts.app')
2
3 @section('content')
4
5     <section>
6         <div class="relative h-screen bg-cover bg-center" style="background-image: url('{{ asset('images/eco-journey/nature-bg.webp') }});">
7             <div class="text-white" style="text-align: center; padding: 10px 0 40px 0;">
8                 <h1>Eco Journey</h1>
9                 <h2>Discover the beauty of nature</h2>
10             </div>
11         </div>
12
13         <div class="text-white" style="text-align: center; padding: 10px 0 40px 0;">
14             <h3>Eco Journey</h3>
15             <h4>Discover the beauty of nature</h4>
16         </div>
17     </section>
18
19     <section id="ecojourney" class="bg-white" style="padding: 10px 0 40px 0;">
20         <div class="text-center" style="padding: 10px 0 40px 0;">
21             <h2>Eco Journey</h2>
22             <h3>Discover the beauty of nature</h3>
23         </div>
24
25         <div class="grid md:grid-cols-3 gap-8 mb-12">
26             <div class="flex flex-col items-center text-center">
27                 
28                 <div class="rounded-[10px] w-full h-auto mb-4">
29                     <p>Eco Journey</p>
30                     <p>Discover the beauty of nature</p>
31                 </div>
32             </div>
33
34             <div class="flex flex-col items-center text-center">
35                 
36                 <div class="rounded-[10px] w-full h-auto mb-4">
37                     <p>Eco Journey</p>
38                     <p>Discover the beauty of nature</p>
39                 </div>
40             </div>
41
42             <div class="flex flex-col items-center text-center">
43                 
44                 <div class="rounded-[10px] w-full h-auto mb-4">
45                     <p>Eco Journey</p>
46                     <p>Discover the beauty of nature</p>
47                 </div>
48             </div>
49         </div>
50
51         <div class="text-center" style="padding: 10px 0 40px 0;">
52             <h3>Eco Journey</h3>
53             <h4>Discover the beauty of nature</h4>
54         </div>
55     </section>
```



```

1  # python3 build.py
2
3  from math import pi, sin, cos, tan, sqrt, exp, log, exp2, log2, exp10, log10, expm1, logm1, expm2, logm2, expm3, logm3, expm4, logm4, expm5, logm5, expm6, logm6, expm7, logm7, expm8, logm8, expm9, logm9, expm10, logm10, expm11, logm11, expm12, logm12, expm13, logm13, expm14, logm14, expm15, logm15, expm16, logm16, expm17, logm17, expm18, logm18, expm19, logm19, expm20, logm20, expm21, logm21, expm22, logm22, expm23, logm23, expm24, logm24, expm25, logm25, expm26, logm26, expm27, logm27, expm28, logm28, expm29, logm29, expm30, logm30, expm31, logm31, expm32, logm32, expm33, logm33, expm34, logm34, expm35, logm35, expm36, logm36, expm37, logm37, expm38, logm38, expm39, logm39, expm40, logm40, expm41, logm41, expm42, logm42, expm43, logm43, expm44, logm44, expm45, logm45, expm46, logm46, expm47, logm47, expm48, logm48, expm49, logm49, expm50, logm50, expm51, logm51, expm52, logm52, expm53, logm53, expm54, logm54, expm55, logm55, expm56, logm56, expm57, logm57, expm58, logm58, expm59, logm59, expm60, logm60, expm61, logm61, expm62, logm62, expm63, logm63, expm64, logm64, expm65, logm65, expm66, logm66, expm67, logm67, expm68, logm68, expm69, logm69, expm70, logm70, expm71, logm71, expm72, logm72, expm73, logm73, expm74, logm74, expm75, logm75, expm76, logm76, expm77, logm77, expm78, logm78, expm79, logm79, expm80, logm80, expm81, logm81, expm82, logm82, expm83, logm83, expm84, logm84, expm85, logm85, expm86, logm86, expm87, logm87, expm88, logm88, expm89, logm89, expm90, logm90, expm91, logm91, expm92, logm92, expm93, logm93, expm94, logm94, expm95, logm95, expm96, logm96, expm97, logm97, expm98, logm98, expm99, logm99, expm100, logm100, expm101, logm101, expm102, logm102, expm103, logm103, expm104, logm104, expm105, logm105, expm106, logm106, expm107, logm107, expm108, logm108, expm109, logm109, expm110, logm110, expm111, logm111, expm112, logm112, expm113, logm113, expm114, logm114, expm115, logm115, expm116, logm116, expm117, logm117, expm118, logm118, expm119, logm119, expm120, logm120, expm121, logm121, expm122, logm122, expm123, logm123, expm124, logm124, expm125, logm125, expm126, logm126, expm127, logm127, expm128, logm128, expm129, logm129, expm130, logm130, expm131, logm131, expm132, logm132, expm133, logm133, expm134, logm134, expm135, logm135, expm136, logm136, expm137, logm137, expm138, logm138, expm139, logm139, expm140, logm140, expm141, logm141, expm142, logm142, expm143, logm143, expm144, logm144, expm145, logm145, expm146, logm146, expm147, logm147, expm148, logm148, expm149, logm149, expm150, logm150, expm151, logm151, expm152, logm152, expm153, logm153, expm154, logm154, expm155, logm155, expm156, logm156, expm157, logm157, expm158, logm158, expm159, logm159, expm160, logm160, expm161, logm161, expm162, logm162, expm163, logm163, expm164, logm164, expm165, logm165, expm166, logm166, expm167, logm167, expm168, logm168, expm169, logm169, expm170, logm170, expm171, logm171, expm172, logm172, expm173, logm173, expm174, logm174, expm175, logm175, expm176, logm176, expm177, logm177, expm178, logm178, expm179, logm179, expm180, logm180, expm181, logm181, expm182, logm182, expm183, logm183, expm184, logm184, expm185, logm185, expm186, logm186, expm187, logm187, expm188, logm188, expm189, logm189, expm190, logm190, expm191, logm191, expm192, logm192, expm193, logm193, expm194, logm194, expm195, logm195, expm196, logm196, expm197, logm197, expm198, logm198, expm199, logm199, expm200, logm200, expm201, logm201, expm202, logm202, expm203, logm203, expm204, logm204, expm205, logm205, expm206, logm206, expm207, logm207, expm208, logm208, expm209, logm209, expm210, logm210, expm211, logm211, expm212, logm212, expm213, logm213, expm214, logm214, expm215, logm215, expm216, logm216, expm217, logm217, expm218, logm218, expm219, logm219, expm220, logm220, expm221, logm221, expm222, logm222, expm223, logm223, expm224, logm224, expm225, logm225, expm226, logm226, expm227, logm227, expm228, logm228, expm229, logm229, expm230, logm230, expm231, logm231, expm232, logm232, expm233, logm233, expm234, logm234, expm235, logm235, expm236, logm236, expm237, logm237, expm238, logm238, expm239, logm239, expm240, logm240, expm241, logm241, expm242, logm242, expm243, logm243, expm244, logm244, expm245, logm245, expm246, logm246, expm247, logm247, expm248, logm248, expm249, logm249, expm250, logm250, expm251, logm251, expm252, logm252, expm253, logm253, expm254, logm254, expm255, logm255, expm256, logm256, expm257, logm257, expm258, logm258, expm259, logm259, expm260, logm260, expm261, logm261, expm262, logm262, expm263, logm263, expm264, logm264, expm265, logm265, expm266, logm266, expm267, logm267, expm268, logm268, expm269, logm269, expm270, logm270, expm271, logm271, expm272, logm272, expm273, logm273, expm274, logm274, expm275, logm275, expm276, logm276, expm277, logm277, expm278, logm278, expm279, logm279, expm280, logm280, expm281, logm281, expm282, logm282, expm283, logm283, expm284, logm284, expm285, logm285, expm286, logm286, expm287, logm287, expm288, logm288, expm289, logm289, expm290, logm290, expm291, logm291, expm292, logm292, expm293, logm293, expm294, logm294, expm295, logm295, expm296, logm296, expm297, logm297, expm298, logm298, expm299, logm299, expm300, logm300, expm301, logm301, expm302, logm302, expm303, logm303, expm304, logm304, expm305, logm305, expm306, logm306, expm307, logm307, expm308, logm308, expm309, logm309, expm310, logm310, expm311, logm311, expm312, logm312, expm313, logm313, expm314, logm314, expm315, logm315, expm316, logm316, expm317, logm317, expm318, logm318, expm319, logm319, expm320, logm320, expm321, logm321, expm322, logm322, expm323, logm323, expm324, logm324, expm325, logm325, expm326, logm326, expm327, logm327, expm328, logm328, expm329, logm329, expm330, logm330, expm331, logm331, expm332, logm332, expm333, logm333, expm334, logm334, expm335, logm335, expm336, logm336, expm337, logm337, expm338, logm338, expm339, logm339, expm340, logm340, expm341, logm341, expm342, logm342, expm343, logm343, expm344, logm344, expm345, logm345, expm346, logm346, expm347, logm347, expm348, logm348, expm349, logm349, expm350, logm350, expm351, logm351, expm352, logm352, expm353, logm353, expm354, logm354, expm355, logm355, expm356,
```

[illegible][illegible]

[illegible]

```

1  #!/usr/bin/perl -w
2
3  use strict;
4  use warnings;
5
6  my $url = "http://www.example.com/";
7  my $user_agent = "Mozilla/5.0 (Windows NT 6.0; rv:2.0.1) Gecko/20100101 Firefox/4.0.1";
8
9  my $response = LWP::Simple::get($url);
10
11  if ($response) {
12      my $content = $response;
13
14      # Extract the content type
15      my $content_type = $content =~ /Content-Type: (.*)/i ? $1 : "text/html";
16
17      # Extract the charset
18      my $charset = $content =~ /charset=([a-z0-9-]+)/i ? $1 : "utf-8";
19
20      # Extract the encoding
21      my $encoding = $content =~ /encoding=([a-z0-9-]+)/i ? $1 : "utf-8";
22
23      # Extract the language
24      my $language = $content =~ /language=([a-z0-9-]+)/i ? $1 : "en";
25
26      # Extract the title
27      my $title = $content =~ /<title>([a-z0-9-]+)/i ? $1 : "Example Title";
28
29      # Extract the meta description
30      my $description = $content =~ /<meta name="description" content="([a-z0-9-]+)/i ? $1 : "Example Description";
31
32      # Extract the meta keywords
33      my $keywords = $content =~ /<meta name="keywords" content="([a-z0-9-]+)/i ? $1 : "Example Keywords";
34
35      # Extract the meta robots
36      my $robots = $content =~ /<meta name="robots" content="([a-z0-9-]+)/i ? $1 : "all";
37
38      # Extract the meta viewport
39      my $viewport = $content =~ /<meta name="viewport" content="([a-z0-9-]+)/i ? $1 : "width=device-width, height=device-height";
40
41      # Extract the meta charset
42      my $meta_charset = $content =~ /<meta charset="([a-z0-9-]+)/i ? $1 : "utf-8";
43
44      # Extract the meta encoding
45      my $meta_encoding = $content =~ /<meta encoding="([a-z0-9-]+)/i ? $1 : "utf-8";
46
47      # Extract the meta language
48      my $meta_language = $content =~ /<meta language="([a-z0-9-]+)/i ? $1 : "en";
49
50      # Extract the meta title
51      my $meta_title = $content =~ /<meta title="([a-z0-9-]+)/i ? $1 : "Example Title";
52
53      # Extract the meta description
54      my $meta_description = $content =~ /<meta description="([a-z0-9-]+)/i ? $1 : "Example Description";
55
56      # Extract the meta keywords
57      my $meta_keywords = $content =~ /<meta keywords="([a-z0-9-]+)/i ? $1 : "Example Keywords";
58
59      # Extract the meta robots
60      my $meta_robots = $content =~ /<meta robots="([a-z0-9-]+)/i ? $1 : "all";
61
62      # Extract the meta viewport
63      my $meta_viewport = $content =~ /<meta viewport="([a-z0-9-]+)/i ? $1 : "width=device-width, height=device-height";
64
65      # Extract the meta charset
66      my $meta_meta_charset = $content =~ /<meta meta-charset="([a-z0-9-]+)/i ? $1 : "utf-8";
67
68      # Extract the meta encoding
69      my $meta_meta_encoding = $content =~ /<meta meta-encoding="([a-z0-9-]+)/i ? $1 : "utf-8";
70
71      # Extract the meta language
72      my $meta_meta_language = $content =~ /<meta meta-language="([a-z0-9-]+)/i ? $1 : "en";
73
74      # Extract the meta title
75      my $meta_meta_title = $content =~ /<meta meta-title="([a-z0-9-]+)/i ? $1 : "Example Title";
76
77      # Extract the meta description
78      my $meta_meta_description = $content =~ /<meta meta-description="([a-z0-9-]+)/i ? $1 : "Example Description";
79
80      # Extract the meta keywords
81      my $meta_meta_keywords = $content =~ /<meta meta-keywords="([a-z0-9-]+)/i ? $1 : "Example Keywords";
82
83      # Extract the meta robots
84      my $meta_meta_robots = $content =~ /<meta meta-robots="([a-z0-9-]+)/i ? $1 : "all";
85
86      # Extract the meta viewport
87      my $meta_meta_viewport = $content =~ /<meta meta-viewport="([a-z0-9-]+)/i ? $1 : "width=device-width, height=device-height";
88
89      # Extract the meta charset
90      my $meta_meta_meta_charset = $content =~ /<meta meta-meta-charset="([a-z0-9-]+)/i ? $1 : "utf-8";
91
92      # Extract the meta encoding
93      my $meta_meta_meta_encoding = $content =~ /<meta meta-meta-encoding="([a-z0-9-]+)/i ? $1 : "utf-8";
94
95      # Extract the meta language
96      my $meta_meta_meta_language = $content =~ /<meta meta-meta-language="([a-z0-9-]+)/i ? $1 : "en";
97
98      # Extract the meta title
99      my $meta_meta_meta_title = $content =~ /<meta meta-meta-title="([a-z0-9-]+)/i ? $1 : "Example Title";
100
101      # Extract the meta description
102      my $meta_meta_meta_description = $content =~ /<meta meta-meta-description="([a-z0-9-]+)/i ? $1 : "Example Description";
103
104      # Extract the meta keywords
105      my $meta_meta_meta_keywords = $content =~ /<meta meta-meta-keywords="([a-z0-9-]+)/i ? $1 : "Example Keywords";
106
107      # Extract the meta robots
108      my $meta_meta_meta_robots = $content =~ /<meta meta-meta-robots="([a-z0-9-]+)/i ? $1 : "all";
109
110      # Extract the meta viewport
111      my $meta_meta_meta_viewport = $content =~ /<meta meta-meta-viewport="([a-z0-9-]+)/i ? $1 : "width=device-width, height=device-height";
112
113      # Extract the meta charset
114      my $meta_meta_meta_meta_charset = $content =~ /<meta meta-meta-meta-charset="([a-z0-9-]+)/i ? $1 : "utf-8";
115
116      # Extract the meta encoding
117      my $meta_meta_meta_meta_encoding = $content =~ /<meta meta-meta-meta-encoding="([a-z0-9-]+)/i ? $1 : "utf-8";
118
119      # Extract the meta language
120      my $meta_meta_meta_meta_language = $content =~ /<meta meta-meta-meta-language="([a-z0-9-]+)/i ? $1 : "en";
121
122      # Extract the meta title
123      my $meta_meta_meta_meta_title = $content =~ /<meta meta-meta-meta-title="([a-z0-9-]+)/i ? $1 : "Example Title";
124
125      # Extract the meta description
126      my $meta_meta_meta_meta_description = $content =~ /<meta meta-meta-meta-description="([a-z0-9-]+)/i ? $1 : "Example Description";
127
128      # Extract the meta keywords
129      my $meta_meta_meta_meta_keywords = $content =~ /<meta meta-meta-meta-keywords="([a-z0-9-]+)/i ? $1 : "Example Keywords";
130
131      # Extract the meta robots
132      my $meta_meta_meta_meta_robots = $content =~ /<meta meta-meta-meta-robots="([a-z0-9-]+)/i ? $1 : "all";
133
134      # Extract the meta viewport
135      my $meta_meta_meta_meta_viewport = $content =~ /<meta meta-meta-meta-viewport="([a-z0-9-]+)/i ? $1 : "width=device-width, height=device-height";
136
137      # Extract the meta charset
138      my $meta_meta_meta_meta_meta_charset = $content =~ /<meta meta-meta-meta-meta-charset="([a-z0-9-]+)/i ? $1 : "utf-8";
139
140      # Extract the meta encoding
141      my $meta_meta_meta_meta_meta_encoding = $content =~ /<meta meta-meta-meta-meta-encoding="([a-z0-9-]+)/i ? $1 : "utf-8";
142
143      # Extract the meta language
144      my $meta_meta_meta_meta_meta_language = $content =~ /<meta meta-meta-meta-meta-language="([a-z0-9-]+)/i ? $1 : "en";
145
146      # Extract the meta title
147      my $meta_meta_meta_meta_meta_title = $content =~ /<meta meta-meta-meta-meta-title="([a-z0-9-]+)/i ? $1 : "Example Title";
148
149      # Extract the meta description
150      my $meta_meta_meta_meta_meta_description = $content =~ /<meta meta-meta-meta-meta-description="([a-z0-9-]+)/i ? $1 : "Example Description";
151
152      # Extract the meta keywords
153      my $meta_meta_meta_meta_meta_keywords = $content =~ /<meta meta-meta-meta-meta-keywords="([a-z0-9-]+)/i ? $1 : "Example Keywords";
154
155      # Extract the meta robots
156      my $meta_meta_meta_meta_meta_robots = $content =~ /<meta meta-meta-meta-meta-robots="([a-z0-9-]+)/i ? $1 : "all";
157
158      # Extract the meta viewport
159      my $meta_meta_meta_meta_meta_viewport = $content =~ /<meta meta-meta-meta-meta-viewport="([a-z0-9-]+)/i ? $1 : "width=device-width, height=device-height";
160
161      # Extract the meta charset
162      my $meta_meta_meta_meta_meta_meta_charset = $content =~ /<meta meta-meta-meta-meta-meta-charset="([a-z0-9-]+)/i ? $1 : "utf-8";
163
164      # Extract the meta encoding
165      my $meta_meta_meta_meta_meta_meta_encoding = $content =~ /<meta meta-meta-meta-meta-meta-encoding="([a-z0-9-]+)/i ? $1 : "utf-8";
166
167      # Extract the meta language
168      my $meta_meta_meta_meta_meta_meta_language = $content =~ /<meta meta-meta-meta-meta-meta-language="([a-z0-9-]+)/i ? $1 : "en";
169
170      # Extract the meta title
171      my $meta_meta_meta_meta_meta_meta_title = $content =~ /<meta meta-meta-meta-meta-meta-title="([a-z0-9-]+)/i ? $1 : "Example Title";
172
173      # Extract the meta description
174      my $meta_meta_meta_meta_meta_meta_description = $content =~ /<meta meta-meta-meta-meta-meta-description="([a-z0-9-]+)/i ? $1 : "Example Description";
175
176      # Extract the meta keywords
177      my $meta_meta_meta_meta_meta_meta_keywords = $content =~ /<meta meta-meta-meta-meta-meta-keywords="([a-z0-9-]+)/i ? $1 : "Example Keywords";
178
179      # Extract the meta robots
180      my $meta_meta_meta_meta_meta_meta_robots = $content =~ /<meta meta-meta-meta-meta-meta-robots="([a-z0-9-]+)/i ? $1 : "all";
181
182      # Extract the meta viewport
183      my $meta_meta_meta_meta_meta_meta_viewport = $content =~ /<meta meta-meta-meta-meta-meta-viewport="([a-z0-9-]+)/i ? $1 : "width=device-width, height=device-height";
184
185      # Extract the meta charset
186      my $meta_meta_meta_meta_meta_meta_meta_charset = $content =~ /<meta meta-meta-meta-meta-meta-meta-charset="([a-z0-9-]+)/i ? $1 : "utf-8";
187
188      # Extract the meta encoding
189      my $meta_meta_meta_meta_meta_meta_meta_encoding = $content =~ /<meta meta-meta-meta-meta-meta-meta-encoding="([a-z0-9-]+)/i ? $1 : "utf-8";
190
191      # Extract the meta language
192      my $meta_meta_meta_meta_meta_meta_meta_language = $content =~ /<meta meta-meta-meta-meta-meta-meta-language="([a-z0-9-]+)/i ? $1 : "en";
193
194      # Extract the meta title
195      my $meta_meta_meta_meta_meta_meta_meta_title = $content =~ /<meta meta-meta-meta-meta-meta-meta-title="([a-z0-9-]+)/i ? $1 : "Example Title";
196
197      # Extract the meta description
198      my $meta_meta_meta_meta_meta_meta_meta_description = $content =~ /<meta meta-meta-meta-meta-meta-meta-description="([a-z0-9-]+)/i ? $1 : "Example Description";
199
200      # Extract the meta keywords
201      my $meta_meta_meta_meta_meta_meta_meta_keywords = $content =~ /<meta meta-meta-meta-meta-meta-meta-keywords="([a-z0-9-]+)/i ? $1 : "Example Keywords";
202
203      # Extract the meta robots
204      my $meta_meta_meta_meta_meta_meta_meta_robots = $content =~ /<meta meta-meta-meta-meta-meta-meta-robots="([a-z0-9-]+)/i ? $1 : "all";
205
206      # Extract the meta viewport
207      my $meta_meta_meta_meta_meta_meta_meta_viewport = $content =~ /<meta meta-meta-meta-meta-meta-meta-viewport="([a-z0-9-]+)/i ? $1 : "width=device-width, height=device-height";
208
209      # Extract the meta charset
210      my $meta_meta_meta_meta_meta_meta_meta_meta_charset = $content =~ /<meta meta-meta-meta-meta-meta-meta-meta-charset="([a-z0-9-]+)/i ? $1 : "utf-8";
211
212      # Extract the meta encoding
213      my $meta_meta_meta_meta_meta_meta_meta_meta_encoding = $content =~ /<meta meta-meta-meta-meta-meta-meta-meta-encoding="([a-z0-9-]+)/i ? $1 : "utf-8";
214
215      # Extract the meta language
216      my $meta_meta_meta_meta_meta_meta_meta_meta_language = $content =~ /<meta meta-meta-meta-meta-meta-meta-meta-language="([a-z0-9-]+)/i ? $1 : "en";
217
218      # Extract the meta title
219      my $meta_meta_meta_meta_meta_meta_meta_meta_title = $content =~ /<meta meta-meta-meta-meta-meta-meta-meta-title="([a-z0-9-]+)/i ? $1 : "Example Title";
220
221      # Extract the meta description
222      my $meta_meta_meta_meta_meta_meta_meta_meta_description = $content =~ /<meta meta-meta-meta-meta-meta-meta-meta-description="([a-z0-9-]+)/i ? $1 : "Example Description";
223
```

```
#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    while (n) {
        int x = n % 10;
        n /= 10;
        if (x == 0) continue;
        cout << x << " ";
    }
    return 0;
}
```

```
Story.php X
50
51 WindowReflector [Explain] [Generate Function Comment] X
52 public function save()
53 {
54     $this->files = $this->tempfiles; // Assign tempfiles to files for validation
55     $this->validate([
56         'title' => 'required|string|max:255',
57         'description' => 'required|string',
58         'files' => 'required|array|min:1|max:3',
59         'files.*' => 'required|file|mimes:jpg,png,gif,webp|max:6000',
60     ]);
61     $user = Auth::user();
62     $post = $user->posts()->create([
63         'title' => $this->title,
64         'description' => $this->description,
65     ]);
66     foreach ($this->tempfiles as $file) {
67         $type = str_starts_with($file->getMimeType(), 'video') ? 'video' : 'image';
68         $path = $file->store('posts/' . $post->id . '/media', 'public');
69         $post->media()->create([
70             'file_path' => $path,
71             'type' => $type,
72         ]);
73     }
74     $this->resetForm();
75     $this->showSuccessModal = true;
76     $this->dispatch('postAdded');
77 }
78
79 WindowReflector [Explain] [Generate Function Comment] X
80 public function resetForm()
81 {
82     $this->title = '';
83     $this->description = '';
84     $this->files = [];
85     $this->tempfiles = [];
86 }
87
88 WindowReflector [Explain] [Generate Function Comment] X
89 public function render()
90 {
91     $userProfile = Auth::user();
92     return view('livewire.profile.story', [
93         'userProfile' => $userProfile
94     ]);
95 }
96
97
```

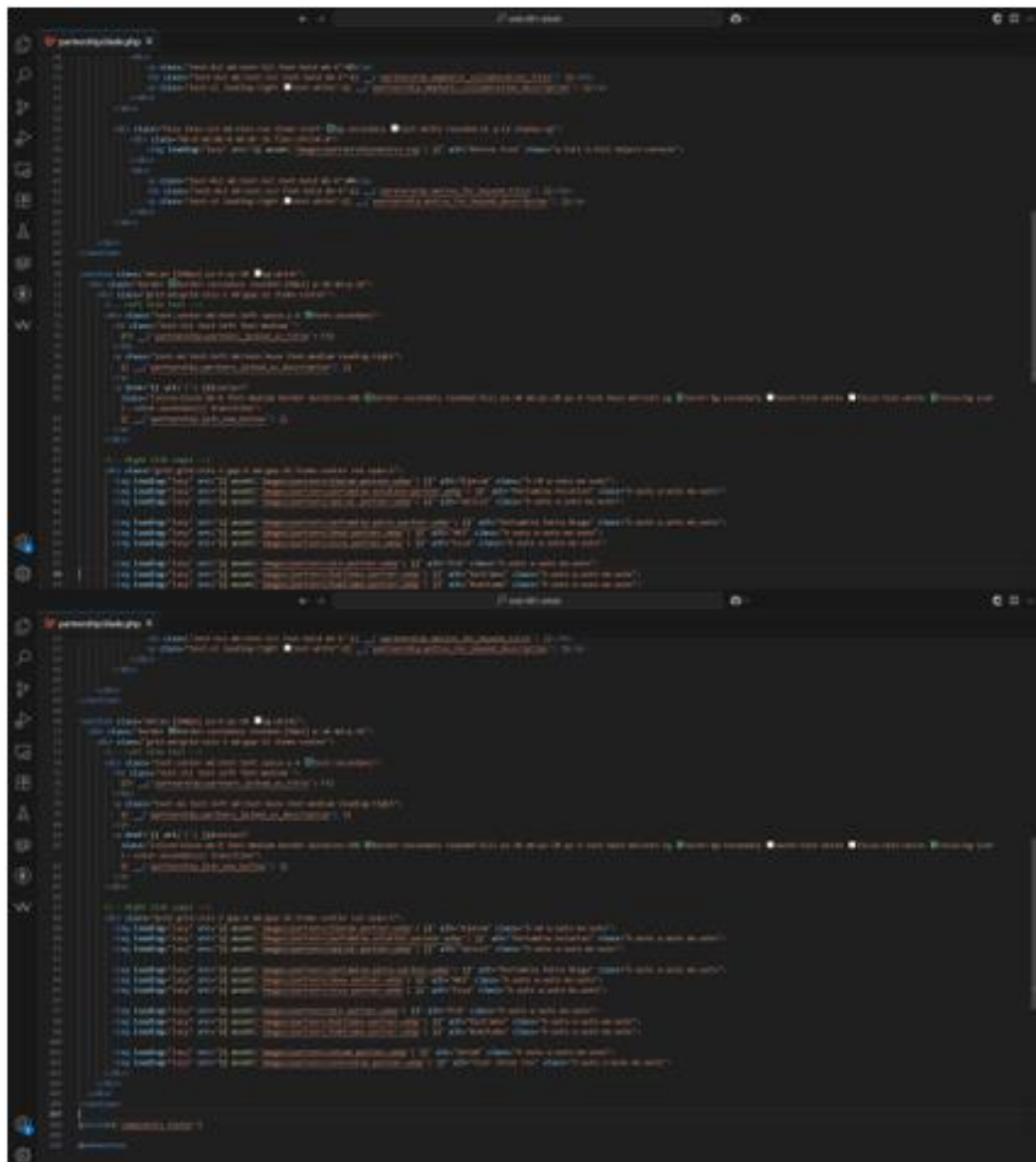
Lampiran 15. Halaman *Ecocircles*

```
eco-circles.blade.php X
1
2 @extends('layouts.app')
3
4 @section('content')
5     <div class="relative h-screen md:px:100px px:4 flex items-center lg:cover lg:center text-white"
6         <div style="background-image: url('{{ asset('images/ecocircles/ecocircles-bg.webp') }})";>
7         @include('components.header')
8         <div class="max-w-xl p:10">
9             <div class="text-3xl md:text-3xl font-bold mb-4">{{ __("ecocircles.hero_title") }}</div>
10             <div class="text-lg max-w-md leading-tight">
11                 {{ __("ecocircles.hero_description") }}
12             </div>
13             <div class="absolute inset-0 lg:opacity-30"></div>
14         </div>
15
16         <div class="px-4 md:px:100px py:16 text-white space-y-24">
17             <div class="text-secondary">
18                 <a href="{{ route('community') }}">
19                     <div class="border border-secondary px-5 py-3 rounded-full text-md font-medium hover:text-white duration-300 hover:bg-secondary focus:text-white focus:outline-none">
20                         <div class="w-8 h-8 w-20 fill:none stroke-current stroke-width=2 d="M18 191.7 76 0 17 76 7 191.7" />
21                         <div class="stroke-linecap=round stroke-join=round stroke-width=2">
22                             {{ __("ecocircles.back_button") }}
23                         </div>
24                     </div>
25                 </div>
26
27                 <div class="grid md:grid-cols-2 gap-12 px-10 items-center justify-center">
28                     
29
30                     <div class="space-y-6 max-w-xs leading-tight font-medium">
31                         <div class="flex items-start space-x-3">
32                             
33                             <div class="text-secondary">{{ __("ecocircles.list_item_1") }}</div>
34                         </div>
35                         <div class="border-t border-black w-full"></div>
36                         <div class="flex items-start space-x-3">
37                             
38                             <div class="text-secondary">{{ __("ecocircles.list_item_2") }}</div>
39                         </div>
40                         <div class="border-t border-black w-full"></div>
41                         <div class="flex items-start space-x-3">
42                             
43                             <div class="text-secondary">{{ __("ecocircles.list_item_3") }}</div>
44                         </div>
45                     </div>
46
47                     <div class="grid md:grid-cols-2 gap-12 px-10 items-center justify-center">
48                         <div class="space-y-6 flex justify-center text-left">
49                             <div class="space-y-6 w-1/2">
50                                 <div class="text-on text-secondary leading-tight">{{ __("ecocircles.join_us_text_1") }}</div>
51                                 <div class="border-t border-black w-full"></div>
52                             </div>
53                         </div>
54                     </div>
55                 </div>
56             </div>
57         </div>
58     </div>
59 </section>
60
```

```
eco-circles.blade.php X
40 <div class="grid md:grid-cols-2 gap-12 px-10 items-center justify-center">
41 <div class="space-y-6 flex justify-center text-left">
42 <div class="space-y-4 w-1/2">
43 <p class="text-sm text-secondary leading-tight">{{ __('eco-circles.join_us_text_1') }}</p>
44 <div class="border-t border-black w-full"></div>
45 </div>
46 <div class="flex justify-start items-center space-x-2">
47 <div class="flex space-x-2">
48 
49 
50 
51 </div>
52 
53 </div>
54 <div class="border-t border-black w-full"></div>
55 <p class="text-sm text-secondary leading-tight">{{ __('eco-circles.join_us_text_2') }}</p>
56 <div class="border-t border-black w-full"></div>
57 <p class="text-xl font-bold text-secondary mb-0">{{ __('eco-circles.active_users_count') }}</p>
58 <p class="text-sm text-secondary">{{ __('eco-circles.active_users_description') }}</p>
59 </div>
60 </div>
61 
62 </div>
63 <div class="pt-24 text-black">
64 <div class="flex flex-col md:flex-row items-center justify-center md:justify-around space-x-12 group transition-all duration-500">
65 <div class="flex flex-col md:flex-row items-center md:space-x-6 text-ds1 md:text-ds1 font-bold group-hover:text-primary transition-colors duration-500 w-20">
66 <span class="transition-transform duration-500 md:group-hover:translate-x-5 group-hover:translate-x-0 group-focus:translate-x-0">{{ __('eco-circles.lets_join_us') }}</span>
67 <img width="200" height="45" viewBox="0 0 150 45" class="w-30 h-20 transition-transform duration-500 group-hover:translate-x-20 group-focus:translate-x-20 hidden md:block fill-none xmlns="http://www.w3.org/2000/svg">
68 <path d="M7 32.5852 112 32.5181 25 180 112 32.5181 25 7" stroke="currentcolor" stroke-width="1" stroke-linecap="round" stroke-join="round"/>
69 </img>
70 <div class="flex flex-col items-center w-20 md:w-40">
71 <img width="60" height="100" viewBox="0 0 60 100" class="w-20 h-30 transition-transform duration-500 group-hover:translate-y-20 md:group-hover:translate-y-0 md:hidden fill-none"
72 <xmlns="http://www.w3.org/2000/svg">
73 <path d="M18 13 83 30 13 70 83 90 13 127 83 167 18 150 13 110 18 70 13 30 18 13" stroke="currentcolor" stroke-width="2" stroke-linecap="round" stroke-join="round" class="group-hover:stroke-primary group-focus:stroke-primary" />
74 </div>
75 </div>
76 
77 <p class="text-sm text-secondary text-center leading-tight">{{ __('eco-circles.our_community_telegram') }}</p>
78 </div>
79 </div>
80 </section>
81 <include('components.footer')>
82 </section>
```

Lampiran 16. Halaman *Partnership*

```
partnership.blade.php X
1 <@extends('layouts.app')>
2
3 <@section('content')>
4 <div class="h-screen relative bg-cover bg-center text-white md:px-[100px] px-6 flex items-center">
5 <div class="background-image: url('{{ asset('images/partnership/partnership-bg.webp') }}');">
6 </div>
7 <include('components.header')>
8 <div class="absolute inset-0 bg-secondary/40 backdrop-blur-[2px]></div>
9 <div class="relative max-w-2xl space-y-6">
10 <h1 class="text-3xl md:text-5xl mb-15 font-semibold leading-tight">
11 <{{ __('partnership.hero_title') }}>
12 </h1>
13 <div class="text-lg font-medium text-justify leading-tight">
14 <{{ __('partnership.hero_description') }}>
15 </div>
16 </div>
17 </section>
18
19 <section class="py-16 px-4 md:px-[100px] text-white">
20 <div class="text-3xl font-bold text-center mb-15">{{ __('partnership.why_choose_us_title') }}</div>
21 <div class="space-y-8">
22 <div class="flex flex-col md:flex-row items-start">
23 <div class="md:mb-8 md:mr-16 flex-shrink-0">
24 
25 </div>
26 <div>
27 <p class="text-ds1 md:text-5xl font-bold mb-4">{{ __('partnership.long_term_strategy_title') }}</p>
28 <div class="text-ds1 md:text-5xl font-bold mb-4">{{ __('partnership.long_term_strategy_title') }}</div>
29 <p class="text-xl leading-tight text-white">{{ __('partnership.long_term_strategy_description') }}</p>
30 </div>
31 </div>
32 </div>
33 <div class="flex flex-col md:flex-row items-start">
34 <div class="md:mb-8 md:mr-16 flex-shrink-0">
35 
36 </div>
37 <div>
38 <p class="text-ds1 md:text-5xl font-bold mb-4">{{ __('partnership.embracing_agile_growth_title') }}</p>
39 <div class="text-ds1 md:text-5xl font-bold mb-4">{{ __('partnership.embracing_agile_growth_title') }}</div>
40 <p class="text-xl leading-tight text-white">{{ __('partnership.embracing_agile_growth_description') }}</p>
41 </div>
42 </div>
43 </div>
44 <div class="flex flex-col md:flex-row items-start">
45 <div class="md:mb-8 md:mr-16 flex-shrink-0">
46 
47 </div>
48 <div>
49 <p class="text-ds1 md:text-5xl font-bold mb-4">{{ __('partnership.emphatic_collaboration_title') }}</p>
50 <div class="text-ds1 md:text-5xl font-bold mb-4">{{ __('partnership.emphatic_collaboration_title') }}</div>
51 <p class="text-xl leading-tight text-white">{{ __('partnership.emphatic_collaboration_description') }}</p>
52 </div>
53 </div>
```



Lampiran 17. Halaman Quiz

Halaman ini terdiri dari beberapa bagian utama. Pertama, terdapat *hero section* berisi judul dan deskripsi. Selanjutnya, ada bagian menampilkan tahapan kuis, diikuti bagian yang menampilkan tiga *green heroes* terbaik minggu ini. Terakhir, tersedia *call to action section* dengan tautan menuju halaman lain di dalam *website*.

Berikut penjelasan detail fitur *quiz* :

Fitur kuis dibangun menggunakan beberapa komponen Livewire :

- a. *Play.php* : Menangani logika saat pengguna memainkan kuis. Komponen ini memuat pertanyaan dan jawaban kuis, mencatat jawaban pengguna, lalu menyimpannya ke *database*. Setelah menjawab, pengguna diarahkan ke halaman *feedback*. Setelah menyelesaikan seluruh kuis, komponen ini menghitung skor dan waktu pengerjaan, lalu mengarahkan ke halaman hasil.
- b. *Result.php* : Menampilkan hasil percobaan kuis. Komponen ini memuat data percobaan dari *database*, menghitung skor pengguna, serta menampilkan jawaban pengguna dan jawaban yang benar.
- c. *Leaderboard.php* : Menampilkan papan peringkat kuis. Komponen ini memuat 3 pengguna dengan skor tertinggi, menyediakan *filter* berdasarkan bulan atau sepanjang waktu, serta fitur pencarian untuk menemukan pengguna tertentu.
- d. *Feedback.php* : Menampilkan umpan balik pertanyaan kuis, jawaban pengguna, jawaban yang benar, dan penjelasan tambahan. Tersedia opsi melanjutkan pertanyaan berikutnya atau menyelesaikan kuis.

```

1  #!/usr/bin/perl
2
3  use strict;
4  use warnings;
5
6  my $url = "http://www.example.com/";
7  my $user_agent = "Mozilla/5.0 (Windows NT 6.0; rv:2.0) Gecko/20100101 Firefox/2.0";
8
9  my $response = LWP::UserAgent->new($user_agent)->get($url);
10
11  my $html = $response->content;
12
13  # Extract the title
14  my $title = $html =~ /<title>(.*)</title>/;
15
16  # Extract the meta description
17  my $meta_desc = $html =~ /<meta name="description" content="(.*)</meta>/;
18
19  # Extract the meta keywords
20  my $meta_keywords = $html =~ /<meta name="keywords" content="(.*)</meta>/;
21
22  # Extract the body text
23  my $body_text = $html =~ /<div class="content">(.*)</div>/;
24
25  # Print the results
26  print "Title: $title\n";
27  print "Meta Description: $meta_desc\n";
28  print "Meta Keywords: $meta_keywords\n";
29  print "Body Text: $body_text\n";
30
31  # Close the connection
32  $response->close;
33
34  # End of script
35  __END__
36
37  #!/usr/bin/perl
38
39  use strict;
40  use warnings;
41
42  my $url = "http://www.example.com/";
43  my $user_agent = "Mozilla/5.0 (Windows NT 6.0; rv:2.0) Gecko/20100101 Firefox/2.0";
44
45  my $response = LWP::UserAgent->new($user_agent)->get($url);
46
47  my $html = $response->content;
48
49  # Extract the title
50  my $title = $html =~ /<title>(.*)</title>/;
51
52  # Extract the meta description
53  my $meta_desc = $html =~ /<meta name="description" content="(.*)</meta>/;
54
55  # Extract the meta keywords
56  my $meta_keywords = $html =~ /<meta name="keywords" content="(.*)</meta>/;
57
58  # Extract the body text
59  my $body_text = $html =~ /<div class="content">(.*)</div>/;
60
61  # Print the results
62  print "Title: $title\n";
63  print "Meta Description: $meta_desc\n";
64  print "Meta Keywords: $meta_keywords\n";
65  print "Body Text: $body_text\n";
66
67  # Close the connection
68  $response->close;
69
70  # End of script
71  __END__

```

```

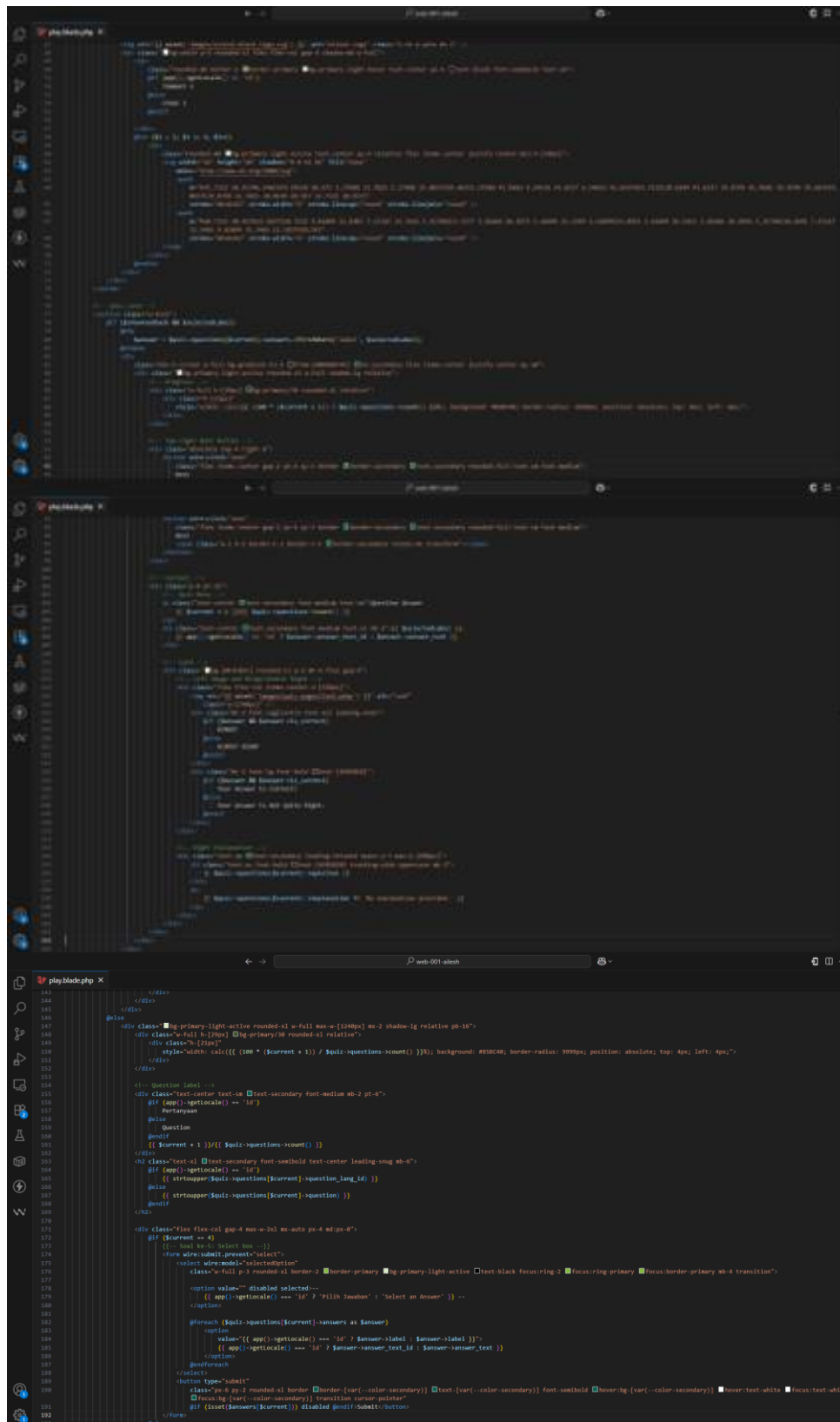
1  #!/usr/bin/perl
2
3  use strict;
4  use warnings;
5
6  my $url = "http://www.example.com/";
7  my $user_agent = "Mozilla/5.0 (Windows NT 6.0; rv:2.0) Gecko/20100101 Firefox/2.0";
8
9  my $response = LWP::UserAgent->new($user_agent)->get($url);
10
11  my $html = $response->content;
12
13  # Extract the title
14  my $title = $html =~ /<title>(.*)</title>/;
15
16  # Extract the meta description
17  my $meta_desc = $html =~ /<meta name="description" content="(.*)</meta>/;
18
19  # Extract the meta keywords
20  my $meta_keywords = $html =~ /<meta name="keywords" content="(.*)</meta>/;
21
22  # Extract the body text
23  my $body_text = $html =~ /<div class="content">(.*)</div>/;
24
25  # Print the results
26  print "Title: $title\n";
27  print "Meta Description: $meta_desc\n";
28  print "Meta Keywords: $meta_keywords\n";
29  print "Body Text: $body_text\n";
30
31  # Close the connection
32  $response->close;
33
34  # End of script
35  __END__
36
37  #!/usr/bin/perl
38
39  use strict;
40  use warnings;
41
42  my $url = "http://www.example.com/";
43  my $user_agent = "Mozilla/5.0 (Windows NT 6.0; rv:2.0) Gecko/20100101 Firefox/2.0";
44
45  my $response = LWP::UserAgent->new($user_agent)->get($url);
46
47  my $html = $response->content;
48
49  # Extract the title
50  my $title = $html =~ /<title>(.*)</title>/;
51
52  # Extract the meta description
53  my $meta_desc = $html =~ /<meta name="description" content="(.*)</meta>/;
54
55  # Extract the meta keywords
56  my $meta_keywords = $html =~ /<meta name="keywords" content="(.*)</meta>/;
57
58  # Extract the body text
59  my $body_text = $html =~ /<div class="content">(.*)</div>/;
60
61  # Print the results
62  print "Title: $title\n";
63  print "Meta Description: $meta_desc\n";
64  print "Meta Keywords: $meta_keywords\n";
65  print "Body Text: $body_text\n";
66
67  # Close the connection
68  $response->close;
69
70  # End of script
71  __END__

```

```

1  #!/usr/bin/perl
2
3  use strict;
4  use warnings;
5
6  my $url = "http://www.example.com/";
7  my $user_agent = "Mozilla/5.0 (Windows NT 6.0; rv:2.0) Gecko/20100101 Firefox/2.0";
8
9  my $response = LWP::UserAgent->new($user_agent)->get($url);
10
11  my $html = $response->content;
12
13  # Extract the title
14  my $title = $html =~ /<title>(.*)</title>/;
15
16  # Extract the meta description
17  my $meta_desc = $html =~ /<meta name="description" content="(.*)</meta>/;
18
19  # Extract the meta keywords
20  my $meta_keywords = $html =~ /<meta name="keywords" content="(.*)</meta>/;
21
22  # Extract the body text
23  my $body_text = $html =~ /<div class="content">(.*)</div>/;
24
25  # Print the results
26  print "Title: $title\n";
27  print "Meta Description: $meta_desc\n";
28  print "Meta Keywords: $meta_keywords\n";
29  print "Body Text: $body_text\n";
30
31  # Close the connection
32  $response->close;
33
34  # End of script
35  __END__
36
37  #!/usr/bin/perl
38
39  use strict;
40  use warnings;
41
42  my $url = "http://www.example.com/";
43  my $user_agent = "Mozilla/5.0 (Windows NT 6.0; rv:2.0) Gecko/20100101 Firefox/2.0";
44
45  my $response = LWP::UserAgent->new($user_agent)->get($url);
46
47  my $html = $response->content;
48
49  # Extract the title
50  my $title = $html =~ /<title>(.*)</title>/;
51
52  # Extract the meta description
53  my $meta_desc = $html =~ /<meta name="description" content="(.*)</meta>/;
54
55  # Extract the meta keywords
56  my $meta_keywords = $html =~ /<meta name="keywords" content="(.*)</meta>/;
57
58  # Extract the body text
59  my $body_text = $html =~ /<div class="content">(.*)</div>/;
60
61  # Print the results
62  print "Title: $title\n";
63  print "Meta Description: $meta_desc\n";
64  print "Meta Keywords: $meta_keywords\n";
65  print "Body Text: $body_text\n";
66
67  # Close the connection
68  $response->close;
69
70  # End of script
71  __END__

```

[illegible]

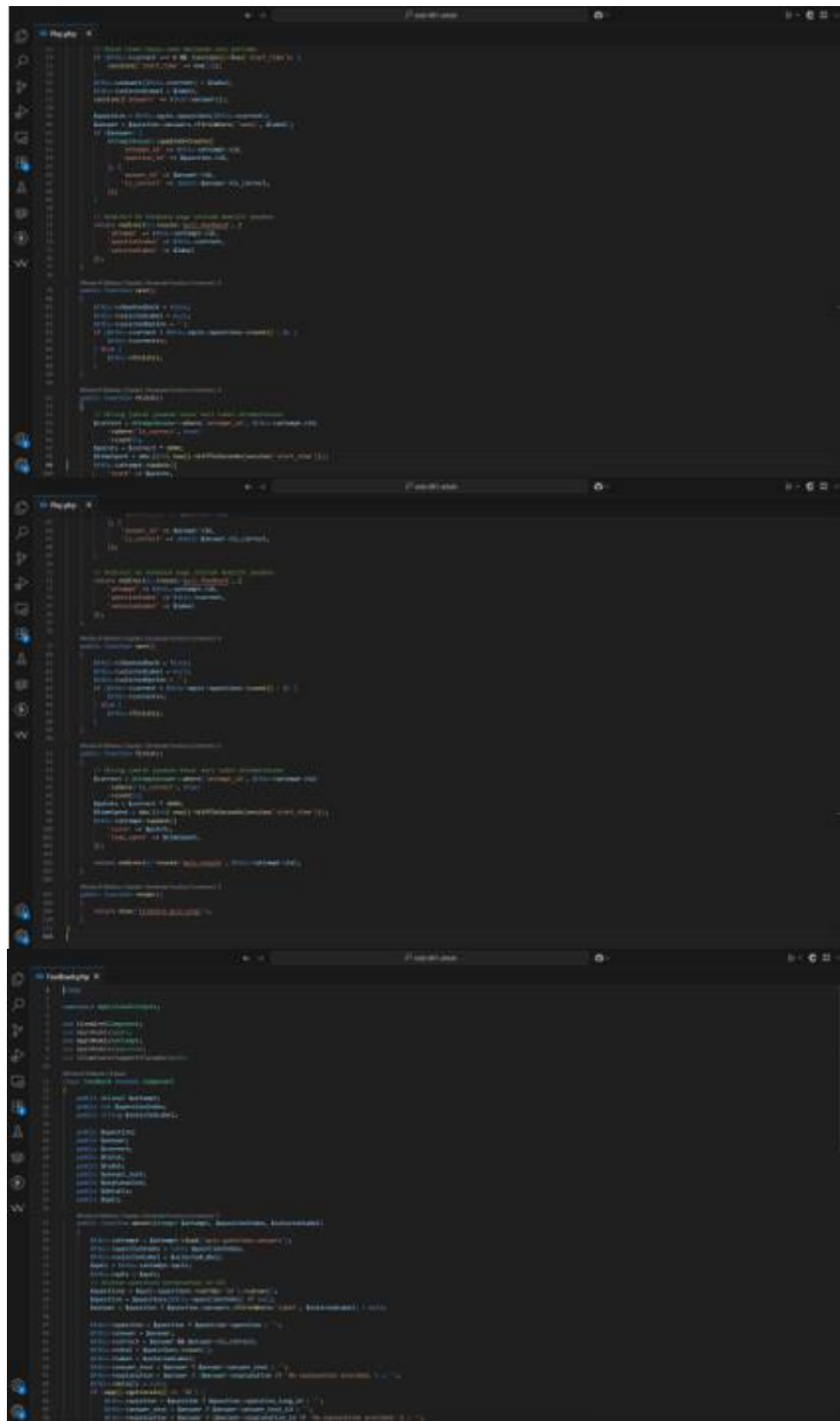
```

1  result-feedback.blade.php X
2
3  <!--
4  -->
5  <div style="width: calc((( 100 * ( $current + 1 ) ) / $quiz-questions-xcount ))%; background: #000000; border-radius: 0000px; position: absolute; top: 4px; left: 4px;"
6  </div>
7
8  <!--
9  -->
10 <div class="p-0 md:p-24 pt-12">
11
12 <!-- Quiz Meta -->
13 <div class="text-center @test-secondary font-medium text-sm">
14 {{ $current-feedback-question-answer-text }}
15
16 {{ $current + 1 }} / {{ $total }}</div>
17 <div class="text-center @test-secondary font-medium text-sm @4 md-6">{{ $label }}.
18 {{ $answer-text }}</div>
19
20 <!-- Card -->
21 @if ($correct)
22 <div
23 class="bg-12 @lg-red-600 rounded-[10px] @text-white text-center flex items-center justify-center text-sm md:text-lg font-sans-serif",
24 {{ $quiz-answers-$id }}</div>
25
26 <!--
27 -->
28 <div class="bg-primary-light-hover rounded-xl p-4 md:p-6 flex flex-col md:flex-row gap-4 md:gap-6">
29
30 <!-- Left: Image and Right -->
31 @if ($correct)
32 <div class="flex flex-col items-center w-full md:w-[300px]"
33 
35 <div class="p-4 text-lg font-bold @text-black">
36 {{ $current-feedback-correct-answer-message }}
37 </div>
38 </div>
39
40 <!--
41 -->
42 <div class="text-sm @text-secondary leading-relaxed space-y-3 w-full @4 md-6">
43 <div class="text-base text-center font-sans-serif @test-secondary tracking-wide uppercase md-2">
44 {{ $question }}
45 </div>
46 <div>
47 {{ $html_entity_decode($explanation) }}
48 </div>
49 <div>
50 {{ $details }}
51 </div>
52 <div>
53 {{ $details }}
54 </div>
55 </div>
56
57 <!--
58 -->
59 </div>
60
61 <!--
62 -->
63 </div>
64
65 <!--
66 -->
67 </div>
68
69 <!--
70 -->
71 </div>
72
73 <!--
74 -->
75 </div>
76
77 <!--
78 -->
79 </div>
80
81 <!--
82 -->
83 </div>
84
85 <!--
86 -->
87 </div>
88
89 <!--
90 -->
91 </div>
92
93 <!--
94 -->
95 </div>
96
97 <!--
98 -->
99 </div>
100
101 <!--
102 -->
103 </div>
104
105 <!--
106 -->
107 </div>
108
109 <!--
110 -->
111 </div>
112
113 <!--
114 -->
115 </div>
116
117 <!--
118 -->
119 </div>
120
121 <!--
122 -->
123 </div>
124
125 <!--
126 -->
127 </div>
128
129 <!--
130 -->
131 </div>
132
133 <!--
134 -->
135 </div>
136
137 <!--
138 -->
139 </div>
140
141 <!--
142 -->
143 </div>
144
145 <!--
146 -->
147 </div>
148
149 <!--
150 -->
151 </div>
152
153 <!--
154 -->
155 </div>
156
157 <!--
158 -->
159 </div>
160
161 <!--
162 -->
163 </div>
164
165 <!--
166 -->
167 </div>
168
169 <!--
170 -->
171 </div>
172
173 <!--
174 -->
175 </div>
176
177 <!--
178 -->
179 </div>
180
181 <!--
182 -->
183 </div>
184
185 <!--
186 -->
187 </div>
188
189 <!--
190 -->
191 </div>
192
193 <!--
194 -->
195 </div>
196
197 <!--
198 -->
199 </div>
200
201 <!--
202 -->
203 </div>
204
205 <!--
206 -->
207 </div>
208
209 <!--
210 -->
211 </div>
212
213 <!--
214 -->
215 </div>
216
217 <!--
218 -->
219 </div>
220
221 <!--
222 -->
223 </div>
224
225 <!--
226 -->
227 </div>
228
229 <!--
230 -->
231 </div>
232
233 <!--
234 -->
235 </div>
236
237 <!--
238 -->
239 </div>
240
241 <!--
242 -->
243 </div>
244
245 <!--
246 -->
247 </div>
248
249 <!--
250 -->
251 </div>
252
253 <!--
254 -->
255 </div>
256
257 <!--
258 -->
259 </div>
260
261 <!--
262 -->
263 </div>
264
265 <!--
266 -->
267 </div>
268
269 <!--
270 -->
271 </div>
272
273 <!--
274 -->
275 </div>
276
277 <!--
278 -->
279 </div>
280
281 <!--
282 -->
283 </div>
284
285 <!--
286 -->
287 </div>
288
289 <!--
290 -->
291 </div>
292
293 <!--
294 -->
295 </div>
296
297 <!--
298 -->
299 </div>
300
301 <!--
302 -->
303 </div>
304
305 <!--
306 -->
307 </div>
308
309 <!--
310 -->
311 </div>
312
313 <!--
314 -->
315 </div>
316
317 <!--
318 -->
319 </div>
320
321 <!--
322 -->
323 </div>
324
325 <!--
326 -->
327 </div>
328
329 <!--
330 -->
331 </div>
332
333 <!--
334 -->
335 </div>
336
337 <!--
338 -->
339 </div>
340
341 <!--
342 -->
343 </div>
344
345 <!--
346 -->
347 </div>
348
349 <!--
350 -->
351 </div>
352
353 <!--
354 -->
355 </div>
356
357 <!--
358 -->
359 </div>
360
361 <!--
362 -->
363 </div>
364
365 <!--
366 -->
367 </div>
368
369 <!--
370 -->
371 </div>
372
373 <!--
374 -->
375 </div>
376
377 <!--
378 -->
379 </div>
380
381 <!--
382 -->
383 </div>
384
385 <!--
386 -->
387 </div>
388
389 <!--
390 -->
391 </div>
392
393 <!--
394 -->
395 </div>
396
397 <!--
398 -->
399 </div>
400
401 <!--
402 -->
403 </div>
404
405 <!--
406 -->
407 </div>
408
409 <!--
410 -->
411 </div>
412
413 <!--
414 -->
415 </div>
416
417 <!--
418 -->
419 </div>
420
421 <!--
422 -->
423 </div>
424
425 <!--
426 -->
427 </div>
428
429 <!--
430 -->
431 </div>
432
433 <!--
434 -->
435 </div>
436
437 <!--
438 -->
439 </div>
440
441 <!--
442 -->
443 </div>
444
445 <!--
446 -->
447 </div>
448
449 <!--
450 -->
451 </div>
452
453 <!--
454 -->
455 </div>
456
457 <!--
458 -->
459 </div>
460
461 <!--
462 -->
463 </div>
464
465 <!--
466 -->
467 </div>
468
469 <!--
470 -->
471 </div>
472
473 <!--
474 -->
475 </div>
476
477 <!--
478 -->
479 </div>
480
481 <!--
482 -->
483 </div>
484
485 <!--
486 -->
487 </div>
488
489 <!--
490 -->
491 </div>
492
493 <!--
494 -->
495 </div>
496
497 <!--
498 -->
499 </div>
500
501 <!--
502 -->
503 </div>
504
505 <!--
506 -->
507 </div>
508
50
```

```

1 <?php
2 namespace App\Livewire\Quiz;
3
4 use Livewire\Component;
5 use App\Models\Quiz;
6 use App\Models\Question;
7 use App\Models\AttemptAnswer;
8 use Illuminate\Support\Facades\Auth;
9
10 class Quiz extends Component
11 {
12     public Quiz $quiz;
13     public array $answers = [];
14     public bool $showFeedback = false;
15     public string $selectedLabel = null;
16     public int $current = 0;
17     public int $timer = 0;
18     public boolean $attempt = null;
19     public $selectedOption = '';
20
21     #[WireRefresh]
22     public function mount(Quiz $quiz)
23     {
24         $this->$quiz = $quiz->load('questions.answers');
25         // gunakan question berdasarkan id agar agar sinkron dengan feedback
26         $this->$quiz->load('questions.question');
27         // gunakan answers pada setiap question berdasarkan label option
28         foreach ($this->$quiz->questions as $q) {
29             $q->refresh('answers');
30             $q->answers->sortBy('label')->values();
31         }
32         $this->$timer = $this->firstOrCreate(
33             ['user_id' => Auth::id()],
34             ['quiz_id' => $quiz->id]
35         );
36         session()->forget('start_time');
37
38         $current = request()->query('current');
39         if (is_numeric($current) && $current >= 0 && $current < $this->$quiz->questions->count()) {
40             $this->$current = (int) $current;
41         }
42     }
43
44     #[WireRefresh]
45     public function select($label = null)
46     {
47         // Untuk soal ke-3 (index 4), gunakan selectedOption jika label kosong
48         if ($this->$current == 4 && empty($label)) {
49             $label = $this->$selectedOption;
50         }
51         if ($label) {
52             return;
53         }
54     }
55 }

```



```

// Subtask 1: Simple operations on the array
// Input: n, m, a[1..n], b[1..m]
// Output: a[1..n], b[1..m]

// Read input
int n, m;
cin >> n >> m;
vector<int> a(n), b(m);
for (int i = 0; i < n; i++) cin >> a[i];
for (int i = 0; i < m; i++) cin >> b[i];

// Process input
// ... (omitted) ...

// Output
// ... (omitted) ...

// End of program
return 0;
}

```

```

// Subtask 2: Simple operations on the array
// Input: n, m, a[1..n], b[1..m]
// Output: a[1..n], b[1..m]

// Read input
int n, m;
cin >> n >> m;
vector<int> a(n), b(m);
for (int i = 0; i < n; i++) cin >> a[i];
for (int i = 0; i < m; i++) cin >> b[i];

// Process input
// ... (omitted) ...

// Output
// ... (omitted) ...

// End of program
return 0;
}

```

```

// Subtask 3: Simple operations on the array
// Input: n, m, a[1..n], b[1..m]
// Output: a[1..n], b[1..m]

// Read input
int n, m;
cin >> n >> m;
vector<int> a(n), b(m);
for (int i = 0; i < n; i++) cin >> a[i];
for (int i = 0; i < m; i++) cin >> b[i];

// Process input
// ... (omitted) ...

// Output
// ... (omitted) ...

// End of program
return 0;
}

```

```

87 loaderboardblade.php h x"
88 <div class="w-full">
89   <div class="text-sm md:text-on font-sansbold □text-black mt-2">{{ __("leaderboard.achievements_title") }}</div>
90   <div class="flex gap-2 justify-start">
91     <div>
92       class="h-16 md:h-4 md:h-16 relative □primary-light-hover flex justify-center rounded-[20px] outline:2 outline-offset:[-1px] □outline-secondary overflow-hidden">
93         
94       </div>
95       <div class="left-4 md:left-[10px] top-20 md:top-[100px] absolute justify-start □text-black text-sm md:text-base font-normal">
96         {{ __("leaderboard.carbon_novice_badge") }}</div>
97     </div>
98   </div>
99   <p class="text-sm md:text-on □text-secondary mt-2 xl:1">{{ __("leaderboard.more_achievements") }}</p>
100   </div>
101   </div>
102 </div>
103 <div>
104   <div>
105     <div class="mt-10 grid grid-cols-1 gap-4 md:gap-6 w-full">
106       <div class="text-left font-style"></div>
107       <div class="col-span-1 w-full md:w-4">
108         <div class="□border-secondary border-1 rounded-lg shadow-on overflow-x-auto">
109           <div class="flex items-center px-2 md:px-6 py-2 border-bottom □border-secondary">
110             <input type="text" placeholder="{{ __("leaderboard.search_placeholder") }}" />
111             <div class="w-full p-2 rounded-full □light-gray border-1 □border-secondary text-sm md:text-on">
112               <div class="stroke-model change-search"/>
113             </div>
114             <svg xmlns="http://www.w3.org/2000/svg" class="w-4 h-4 md:w:5 md:h:5 □text-black xl:6 md:xl:8" fill="none">
115               <stroke width="2" stroke="black" stroke-linecap="round" stroke-linejoin="round" stroke-width="2">
116                 d="M21 21 4.35 4.35A8 8 0 10 16 8 0 0 0 16 8" />
117             </svg>
118           </div>
119           <div class="divide-y □divide-gray-100 text-sm md:text-on">
120             <div class="leaders as li □leader"></div>
121             <div class="flex items-center px-6 py-2">
122               <div class="w-4 h-4 rounded-full □blue-secondary □text-white text-sm flex items-center justify-center mt-5">
123                 {{ $ i + 1 }}</div>
124               <div class="flex-grow-1 □font-sansbold"></div>
125             </div>
126             <div class="flex-1 font-sansbold"></div>
127             <div class="text-sm">{{ leader_name }}</div>
128             <div class="text-sm □text-black">{{ leader_total_score }}</div>
129           </div>
130         </div>
131       </div>
132     </div>
133     <div class="px-4 py-6 border-top □text-secondary text-sm">{{ __("leaderboard.not_found_message") }}</div>
134   </div>
135 </div>
136 </div>

```

```

1 | //vue
2
3 | namespace App\Livewire\Quiz;
4
5 | use App\Models\Quiz;
6 | use App\Models\User;
7 | use Livewire\Component;
8 | use Illuminate\Support\Facades\DB;
9
10 | WindowReflector { Explan
11 | class Leaderboard extends Component
12 | {
13 |     public Quiz $quiz;
14 |     public $answers = [];
15 |     public $questionIndex = 1;
16 |     public $search = '';
17 |     public $filter = 'month'; // 'month' or 'all'
18
19 |     WindowReflector { Explan { Generate Function Comment } X
20 |     public function mount(Quiz $quiz)
21 |     {
22 |         $this->$quiz = $quiz;
23 |         $this->updateAnswers();
24 |     }
25
26 |     WindowReflector { Explan { Generate Function Comment } X
27 |     public function updateSearch()
28 |     {
29 |     }
30 |
31 |     WindowReflector { Explan { Generate Function Comment } X
32 |     public function setFilter($filter)
33 |     {
34 |         $this->$filter = $filter;
35 |         $this->updateAnswers();
36 |     }
37
38 |     WindowReflector { Explan { Generate Function Comment } X
39 |     public function updateAnswers()
40 |     {
41 |         // Pulling top 5, no search
42 |         $questionsQuery = User::select('users.id', 'users.name', 'users.avatar', DB::raw('SUM(Attempts.score) as total_score'))
43 |             ->join('Attempts', 'Attempts.user_id', 'users.id')
44 |             ->where('Attempts.user_id', $this->$quiz->id)
45 |             ->groupBy('users.id', 'users.name', 'users.avatar')
46 |             ->orderByDesc('total_score')
47 |             ->take(5);
48 |
49 |         if ($this->$filter == 'month') {
50 |             $questionsQuery->whereMonth('Attempts.created_at', now()->month)
51 |                 ->whereYear('Attempts.created_at', now()->year);
52 |         }
53 |
54 |         $this->$questionIndex = $questionsQuery
55 |             ->groupBy('users.id', 'users.name', 'users.avatar')
56 |             ->orderByDesc('total_score')
57 |             ->take(5);
58 |     }
59 | }

```

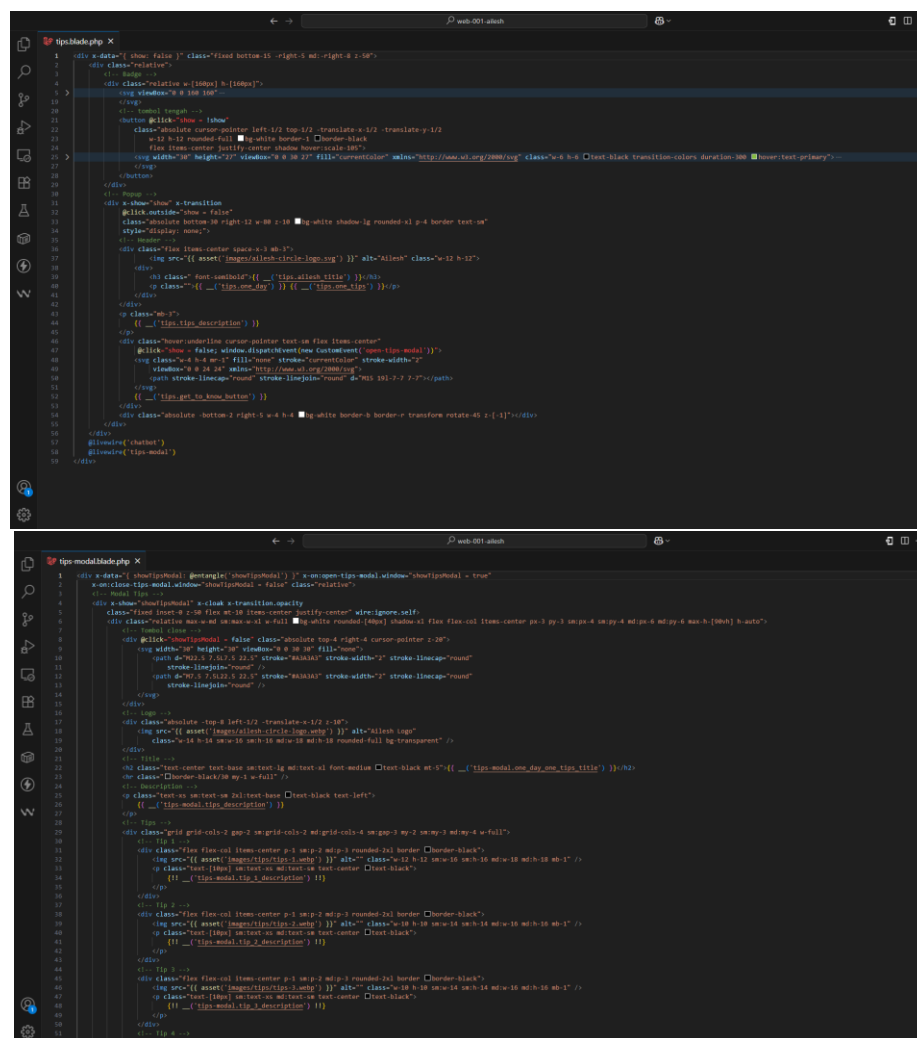
```

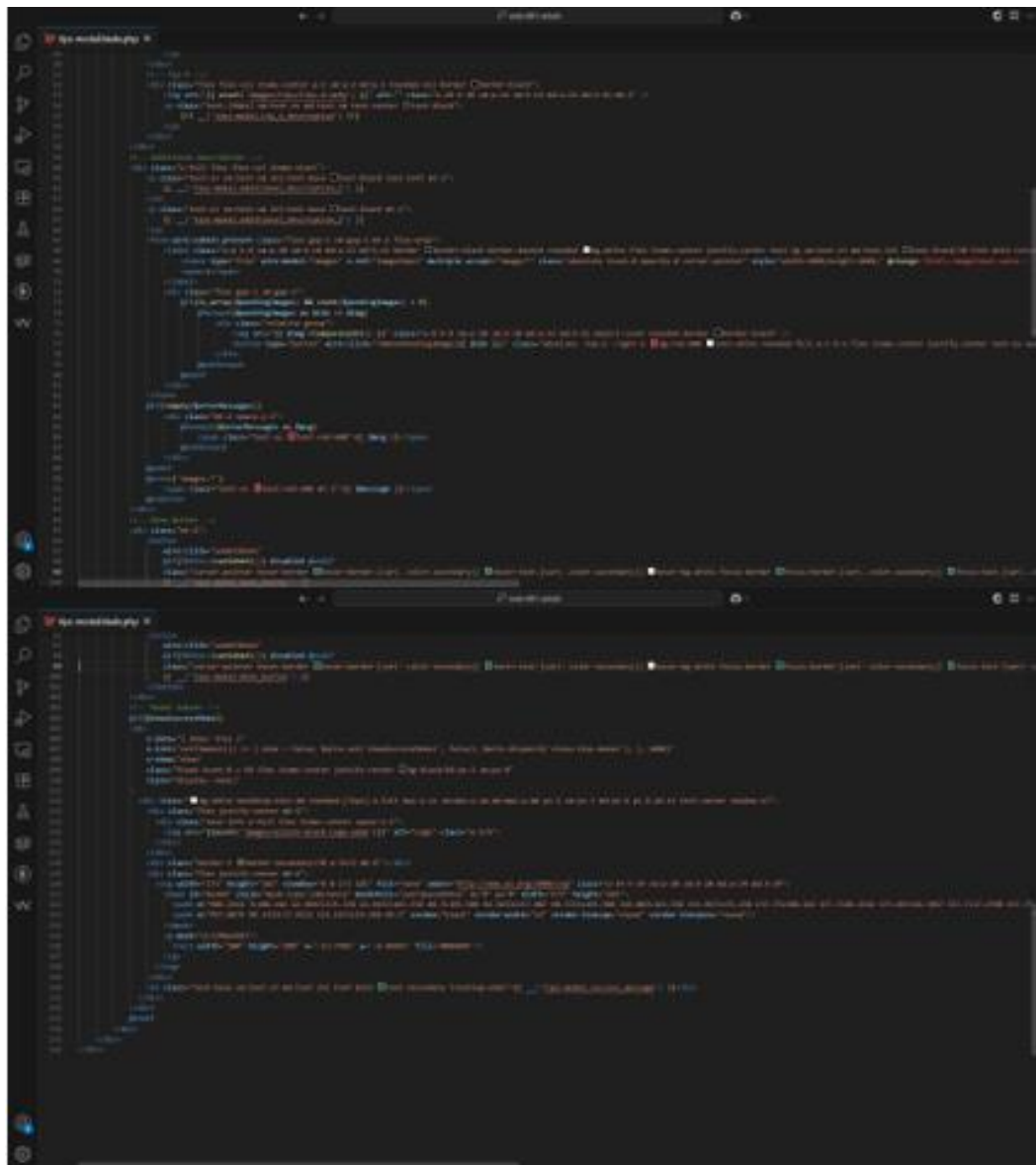
1 // leaderboard.php X
2
3 Window Refactor (Explain) (Generate Function Comment) X
4
5 public function updateLeaders()
6 {
7     // Position: top 5, no search
8     $podiumQuery = mysqli_select('users.id','users.name','users.avatar', DB::raw('SUM(attempts.score) as total_score'))
9     ->join('attempts','attempts.user_id','users.id')
10     ->where('attempts.quiz_id', $this->quiz->id);
11     if ($this->filter == 'month') {
12         $podiumQuery->whereMonth('attempts.created_at', now())->month()
13         ->whereYear('attempts.created_at', now())->year();
14     }
15     $this->podiumLeaders = $podiumQuery
16     ->groupBy('users.id','users.name','users.avatar')
17     ->orderByDesc('total_score')
18     ->take(5)
19     ->get();
20
21     // Ranking table: filter by search
22     $tableQuery = mysqli_select('users.id','users.name','users.avatar', DB::raw('SUM(attempts.score) as total_score'))
23     ->join('attempts','attempts.user_id','users.id')
24     ->where('attempts.quiz_id', $this->quiz->id);
25     if ($this->filter == 'month') {
26         $tableQuery->whereMonth('attempts.created_at', now())->month()
27         ->whereYear('attempts.created_at', now())->year();
28     }
29     if ($this->search) {
30         $separator = DB::getRawExpression();
31         $separator = $separator == '||' ? '||like' : '||like';
32         $tableQuery->where('users.name', $separator, '%' . $this->search . '%');
33     }
34     $this->leaders = $tableQuery
35     ->groupBy('users.id','users.name','users.avatar')
36     ->orderByDesc('total_score')
37     ->take(10)
38     ->get();
39 }
40
41 Window Refactor (Explain) (Generate Function Comment) X
42
43 public function render()
44 {
45     $this->updateLeaders();
46     return view('llmwire.quiz.leaderboard', [
47         'leaders' => $this->leaders,
48         'podiumLeaders' => $this->podiumLeaders,
49         'quiz' => $this->quiz,
50         'search' => $this->search,
51         'filter' => $this->filter,
52     ]);
53 }
54
55

```

Lampiran 18. Fitur *One day one tips*

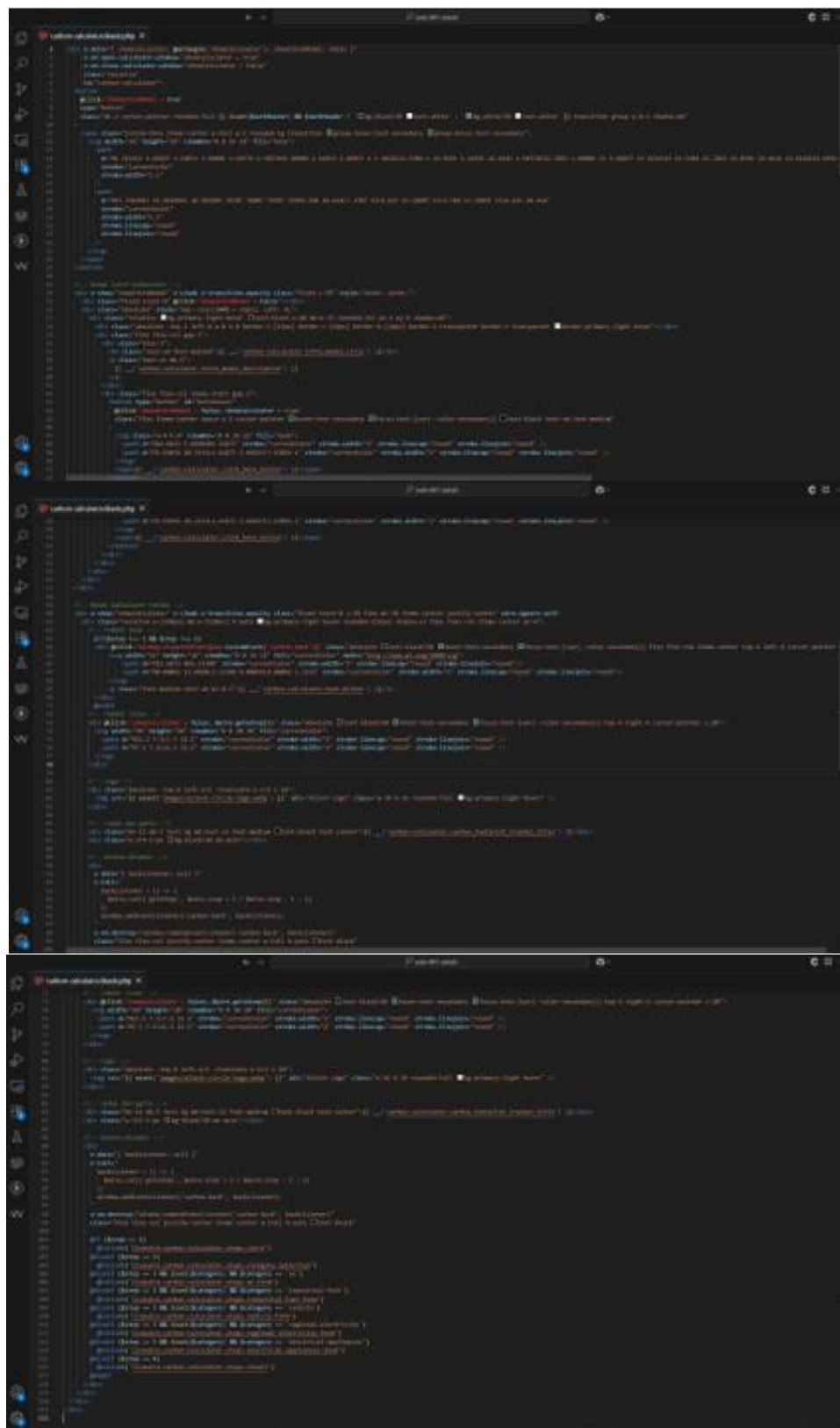
Fitur yang menampilkan satu tips setiap hari kepada pengguna. Fitur ini diimplementasikan menggunakan sebuah modal. Modal tersebut didefinisikan dalam file *app/Livewire/TipsModal.php*. Komponen Livewire ini menangani logika untuk modal, termasuk menampilkan dan menyembunyikan modal, serta mengelola unggahan gambar. Komponen ini memungkinkan pengguna mengunggah hingga tiga gambar, yang kemudian disimpan ke *server*. Selain itu, komponen juga menyertakan validasi untuk gambar yang diunggah, seperti pengecekan ukuran *file* dan jenis *file*.





Lampiran 19. Fitur *Carbon Footprint Tracker*

Fitur yang memungkinkan pengguna menghitung jejak karbon mereka. Fitur ini diimplementasikan menggunakan komponen Livewire app/Livewire/CarbonCalculator.php. Komponen ini menangani logika perhitungan, termasuk kategori perhitungan AC, bahan bakar industri, kendaraan, listrik regional, dan peralatan listrik. Komponen ini menggunakan *session* untuk menyimpan hasil perhitungan, sehingga pengguna dapat melihat total emisi mereka.



[illegible][illegible]

```

1  #!/usr/bin/perl
2
3  use strict;
4  use warnings;
5
6  my $url = "http://www.example.com/";
7  my $user_agent = "Mozilla/5.0 (Windows NT 6.0; rv:2.0) Gecko/20100101 Firefox/2.0";
8
9  my $response = LWP::UserAgent->new( agent => $user_agent )->get($url);
10
11  my $html = $response->content;
12
13  # Parse the HTML
14  my $dom = HTML::Dom->new($html);
15
16  # Find all links
17  my @links = $dom->getElementsByTagName("a");
18
19  # Print the links
20  foreach my $link (@links) {
21      my $href = $link->getAttribute("href");
22      my $text = $link->getText;
23      print "$text ($href)\n";
24  }
25
26  # Close the DOM
27  $dom->close();
28
29  # Close the UserAgent
30  $response->close();
31
32  # End of script
33  exit 0;
34

```

[illegible][illegible]

```

1  #!/usr/bin/perl -w
2
3  use strict;
4  use warnings;
5
6  my $VERSION = "1.0.0";
7  my $AUTHOR = "John Doe";
8  my $COPYRIGHT = "Copyright 2023 John Doe";
9
10 my $SYNOPSIS = "Usage: perl script.pl [options]";
11 my $DESCRIPTION = "This script is a placeholder for a Perl script. It contains various comments and code snippets for documentation purposes.";
12
13 my $USAGE = "
14     perl script.pl [options]
15
16     Options:
17     -h, --help            Display this help message.
18     -v, --version         Display the version number.
19     -o, --output FILENAME Write output to FILENAME.
20     -d, --debug           Enable debug mode.
21     -s, --silent          Suppress output.
22     -f, --force           Force execution.
23     -c, --config FILENAME Load configuration from FILENAME.
24     -e, --exec CMD        Execute CMD instead of the script.
25     -i, --interactive     Enable interactive mode.
26     -q, --quiet           Quiet mode.
27     -n, --no-color        Disable color output.
28     -m, --mode MODE       Set mode to MODE.
29     -p, --port PORT       Set port to PORT.
30     -u, --url URL          Set URL to URL.
31     -t, --timeout SECS    Set timeout to SECS.
32     -w, --width COLS       Set width to COLS.
33     -H, --help-all       Display all help information.
34
35     Examples:
36     perl script.pl -o output.txt
37     perl script.pl -d -s -f
38     perl script.pl -c config.yaml -e 'perl -e print \"Hello World\"'
39     perl script.pl -i -q -n -m read-only -p 8080 -u http://localhost:8080 -t 30 -w 80
40
41     For more information, see the README file.
42 ";
43
44 my $EXAMPLES = "
45     # Example 1: Basic usage
46     perl script.pl
47
48     # Example 2: With options
49     perl script.pl -o output.txt -d -s -f
50
51     # Example 3: With configuration and command
52     perl script.pl -c config.yaml -e 'perl -e print \"Hello World\"'
53
54     # Example 4: With multiple options
55     perl script.pl -i -q -n -m read-only -p 8080 -u http://localhost:8080 -t 30 -w 80
56 ";
57
58 my $README = "
59     This script is a placeholder for a Perl script. It contains various comments and code snippets for documentation purposes.
60
61     The script is designed to be flexible and easy to use. It supports a wide range of options and can be configured to suit your needs.
62
63     For more information, see the README file.
64 ";
65
66 my $LICENSE = "
67     This script is licensed under the MIT License.
68
69     Copyright 2023 John Doe
70
71     Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
72
73     The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
74
75     THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
76 ";
77
78 my $FOOTER = "
79     $SYNOPSIS
80     $DESCRIPTION
81     $USAGE
82     $EXAMPLES
83     $README
84     $LICENSE
85 ";
86
87 # Main function
88 sub main {
89     my $argc = scalar(@ARGV);
90     my $argv = \%ARGV;
91
92     # Parse command line arguments
93     my $options = parse_options($argc, $argv);
94
95     # Execute the script
96     execute_script($options);
97 }
98
99 # Parse command line arguments
100 sub parse_options {
101     my ($argc, $argv) = @_;
102
103     my %options = ();
104
105     for my $i (0..$argc-1) {
106         my $arg = $argv->[$i];
107
108         if ($arg =~ /^--help$/) {
109             print $SYNOPSIS;
110             print $DESCRIPTION;
111             return;
112         }
113
114         if ($arg =~ /^--version$/) {
115             print $VERSION;
116             return;
117         }
118
119         if ($arg =~ /^-o\s+([^ ]+)$/) {
120             $options{output} = $1;
121         }
122
123         if ($arg =~ /^-d$/) {
124             $options{debug} = 1;
125         }
126
127         if ($arg =~ /^-s$/) {
128             $options{silent} = 1;
129         }
130
131         if ($arg =~ /^-f$/) {
132             $options{force} = 1;
133         }
134
135         if ($arg =~ /^-c\s+([^ ]+)$/) {
136             $options{config} = $1;
137         }
138
139         if ($arg =~ /^-e\s+'([^ ]+)'$/) {
140             $options{exec} = $1;
141         }
142
143         if ($arg =~ /^-i$/) {
144             $options{interactive} = 1;
145         }
146
147         if ($arg =~ /^-q$/) {
148             $options{quiet} = 1;
149         }
150
151         if ($arg =~ /^-n$/) {
152             $options{no_color} = 1;
153         }
154
155         if ($arg =~ /^-m\s+([^ ]+)$/) {
156             $options{mode} = $1;
157         }
158
159         if ($arg =~ /^-p\s+([^ ]+)$/) {
160             $options{port} = $1;
161         }
162
163         if ($arg =~ /^-u\s+([^ ]+)$/) {
164             $options{url} = $1;
165         }
166
167         if ($arg =~ /^-t\s+([^ ]+)$/) {
168             $options{timeout} = $1;
169         }
170
171         if ($arg =~ /^-w\s+([^ ]+)$/) {
172             $options{width} = $1;
173         }
174
175         if ($arg =~ /^-H$/) {
176             print $SYNOPSIS;
177             print $DESCRIPTION;
178             print $USAGE;
179             print $EXAMPLES;
180             print $README;
181             print $LICENSE;
182             return;
183         }
184     }
185
186     return %options;
187 }
188
189 # Execute the script
190 sub execute_script {
191     my (%options) = @_;
192
193     # TODO: Implement the script logic here
194 }
195
196 # Main execution
197 main();

```

[illegible][illegible]

```

1  #!/usr/bin/perl -w
2
3  use strict;
4  use warnings;
5
6  my $script = $0;
7  my $version = "1.0.0";
8
9  my $help = "
10  Usage: $script [options]
11  Options:
12  -h, --help            display this help message
13  -v, --version         display version information
14  -f, --file FILE        input file
15  -o, --output FILE      output file
16  -d, --debug           enable debugging
17  -s, --size SIZE        file size in bytes
18  -t, --type TYPE        file type
19  -m, --mode MODE        file mode
20  -c, --checksum         calculate checksum
21  -C, --checksum FILE    checksum file
22  -M, --mtime MTIME      modification time
23  -A, --atime ATIME      access time
24  -U, --uid UID           user ID
25  -G, --gid GID           group ID
26  -L, --links LINKS       number of links
27  -P, --perms PERMS       permissions
28  -S, --selinux SELINUX   SELinux context
29  -Z, --zfs ZFS           ZFS dataset name
30  -B, --blocksize BLOCKSIZE block size
31  -b, --blocksize64 BLOCKSIZE64 block size in 64-bit
32  -B64, --blocksize6464 BLOCKSIZE6464 block size in 64-bit
33  -B128, --blocksize128 BLOCKSIZE128 block size in 128-bit
34  -B256, --blocksize256 BLOCKSIZE256 block size in 256-bit
35  -B512, --blocksize512 BLOCKSIZE512 block size in 512-bit
36  -B1024, --blocksize1024 BLOCKSIZE1024 block size in 1024-bit
37  -B2048, --blocksize2048 BLOCKSIZE2048 block size in 2048-bit
38  -B4096, --blocksize4096 BLOCKSIZE4096 block size in 4096-bit
39  -B8192, --blocksize8192 BLOCKSIZE8192 block size in 8192-bit
40  -B16384, --blocksize16384 BLOCKSIZE16384 block size in 16384-bit
41  -B32768, --blocksize32768 BLOCKSIZE32768 block size in 32768-bit
42  -B65536, --blocksize65536 BLOCKSIZE65536 block size in 65536-bit
43  -B131072, --blocksize131072 BLOCKSIZE131072 block size in 131072-bit
44  -B262144, --blocksize262144 BLOCKSIZE262144 block size in 262144-bit
45  -B524288, --blocksize524288 BLOCKSIZE524288 block size in 524288-bit
46  -B1048576, --blocksize1048576 BLOCKSIZE1048576 block size in 1048576-bit
47  -B2097152, --blocksize2097152 BLOCKSIZE2097152 block size in 2097152-bit
48  -B4194304, --blocksize4194304 BLOCKSIZE4194304 block size in 4194304-bit
49  -B8388608, --blocksize8388608 BLOCKSIZE8388608 block size in 8388608-bit
50  -B16777216, --blocksize16777216 BLOCKSIZE16777216 block size in 16777216-bit
51  -B33554432, --blocksize33554432 BLOCKSIZE33554432 block size in 33554432-bit
52  -B67108864, --blocksize67108864 BLOCKSIZE67108864 block size in 67108864-bit
53  -B134217728, --blocksize134217728 BLOCKSIZE134217728 block size in 134217728-bit
54  -B268435456, --blocksize268435456 BLOCKSIZE268435456 block size in 268435456-bit
55  -B536870912, --blocksize536870912 BLOCKSIZE536870912 block size in 536870912-bit
56  -B1073741824, --blocksize1073741824 BLOCKSIZE1073741824 block size in 1073741824-bit
57  -B2147483648, --blocksize2147483648 BLOCKSIZE2147483648 block size in 2147483648-bit
58  -B4294967296, --blocksize4294967296 BLOCKSIZE4294967296 block size in 4294967296-bit
59  -B8589934592, --blocksize8589934592 BLOCKSIZE8589934592 block size in 8589934592-bit
60  -B17179869184, --blocksize17179869184 BLOCKSIZE17179869184 block size in 17179869184-bit
61  -B34359738368, --blocksize34359738368 BLOCKSIZE34359738368 block size in 34359738368-bit
62  -B68719476736, --blocksize68719476736 BLOCKSIZE68719476736 block size in 68719476736-bit
63  -B137438953472, --blocksize137438953472 BLOCKSIZE137438953472 block size in 137438953472-bit
64  -B274877906944, --blocksize274877906944 BLOCKSIZE274877906944 block size in 274877906944-bit
65  -B549755813888, --blocksize549755813888 BLOCKSIZE549755813888 block size in 549755813888-bit
66  -B1099511627776, --blocksize1099511627776 BLOCKSIZE1099511627776 block size in 1099511627776-bit
67  -B2199023255552, --blocksize2199023255552 BLOCKSIZE2199023255552 block size in 2199023255552-bit
68  -B4398046511104, --blocksize4398046511104 BLOCKSIZE4398046511104 block size in 4398046511104-bit
69  -B8796093022208, --blocksize8796093022208 BLOCKSIZE8796093022208 block size in 8796093022208-bit
70  -B17592186044416, --blocksize17592186044416 BLOCKSIZE17592186044416 block size in 17592186044416-bit
71  -B35184372088832, --blocksize35184372088832 BLOCKSIZE35184372088832 block size in 35184372088832-bit
72  -B70368744177664, --blocksize70368744177664 BLOCKSIZE70368744177664 block size in 70368744177664-bit
73  -B140737488355328, --blocksize140737488355328 BLOCKSIZE140737488355328 block size in 140737488355328-bit
74  -B281474976710656, --blocksize281474976710656 BLOCKSIZE281474976710656 block size in 281474976710656-bit
75  -B562949953421312, --blocksize562949953421312 BLOCKSIZE562949953421312 block size in 562949953421312-bit
76  -B1125899906842624, --blocksize1125899906842624 BLOCKSIZE1125899906842624 block size in 1125899906842624-bit
77  -B2251799813685248, --blocksize2251799813685248 BLOCKSIZE2251799813685248 block size in 2251799813685248-bit
78  -B4503599627370496, --blocksize4503599627370496 BLOCKSIZE4503599627370496 block size in 4503599627370496-bit
79  -B9007199254740992, --blocksize9007199254740992 BLOCKSIZE9007199254740992 block size in 9007199254740992-bit
80  -B18014398509481984, --blocksize18014398509481984 BLOCKSIZE18014398509481984 block size in 18014398509481984-bit
81  -B36028797018963968, --blocksize36028797018963968 BLOCKSIZE36028797018963968 block size in 36028797018963968-bit
82  -B72057594037927936, --blocksize72057594037927936 BLOCKSIZE72057594037927936 block size in 72057594037927936-bit
83  -B144115188075855872, --blocksize144115188075855872 BLOCKSIZE144115188075855872 block size in 144115188075855872-bit
84  -B288230376151711744, --blocksize288230376151711744 BLOCKSIZE288230376151711744 block size in 288230376151711744-bit
85  -B576460752303423488, --blocksize576460752303423488 BLOCKSIZE576460752303423488 block size in 576460752303423488-bit
86  -B1152921504606846976, --blocksize1152921504606846976 BLOCKSIZE1152921504606846976 block size in 1152921504606846976-bit
87  -B2305843009213693952, --blocksize2305843009213693952 BLOCKSIZE2305843009213693952 block size in 2305843009213693952-bit
88  -B4611686018427387904, --blocksize4611686018427387904 BLOCKSIZE4611686018427387904 block size in 4611686018427387904-bit
89  -B9223372036854775808, --blocksize9223372036854775808 BLOCKSIZE9223372036854775808 block size in 9223372036854775808-bit
90  -B18446744073709551616, --blocksize18446744073709551616 BLOCKSIZE18446744073709551616 block size in 18446744073709551616-bit
91  -B36893488147419103232, --blocksize36893488147419103232 BLOCKSIZE36893488147419103232 block size in 36893488147419103232-bit
92  -B73786976294838
```

```

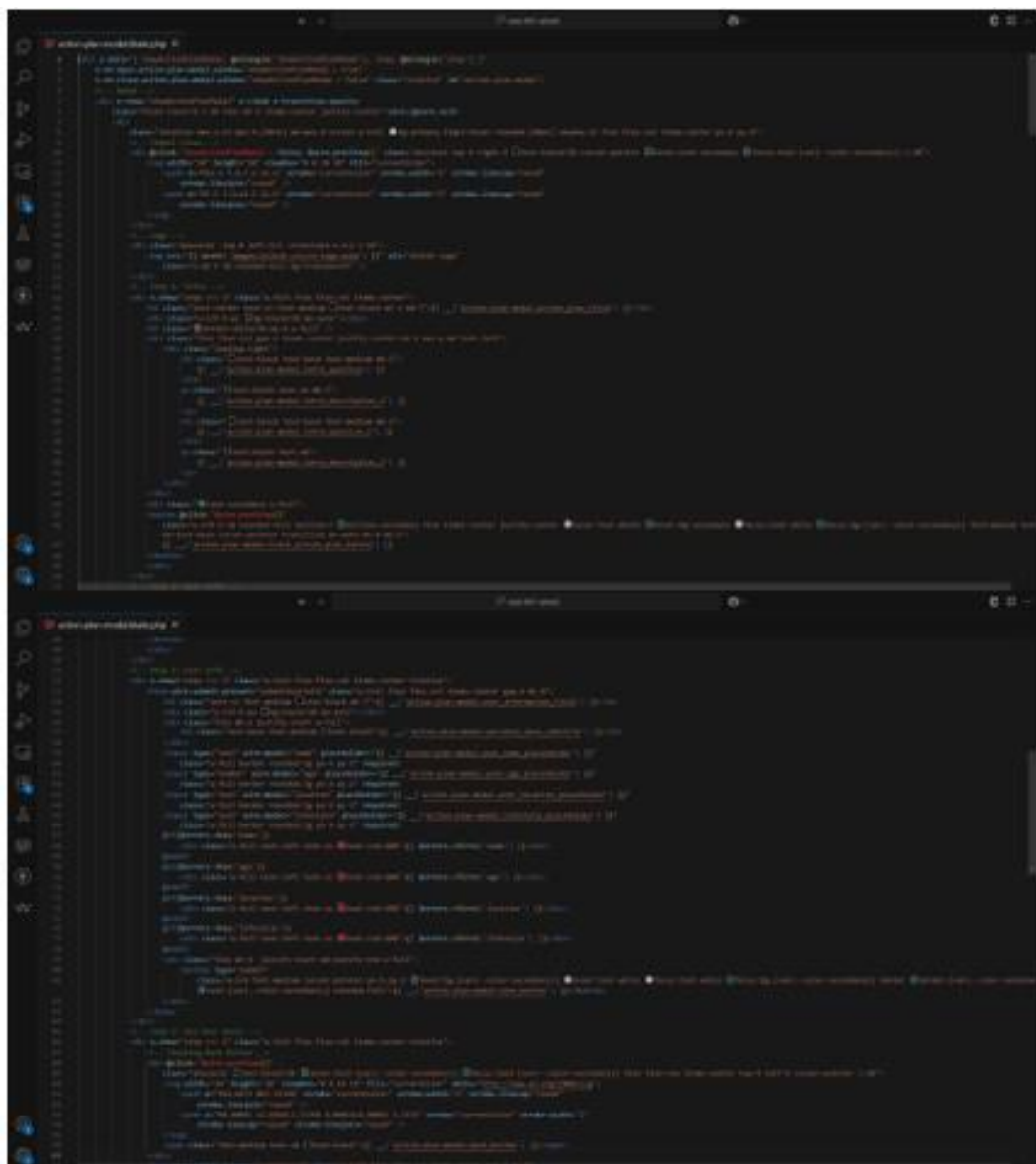
1 100% (source to file) and 100% (source to file)
2 100% (source to file)
3 100% (source to file)
4 100% (source to file)
5 100% (source to file)
6 100% (source to file)
7 100% (source to file)
8 100% (source to file)
9 100% (source to file)
10 100% (source to file)
11 100% (source to file)
12 100% (source to file)
13 100% (source to file)
14 100% (source to file)
15 100% (source to file)
16 100% (source to file)
17 100% (source to file)
18 100% (source to file)
19 100% (source to file)
20 100% (source to file)
21 100% (source to file)
22 100% (source to file)
23 100% (source to file)
24 100% (source to file)
25 100% (source to file)
26 100% (source to file)
27 100% (source to file)
28 100% (source to file)
29 100% (source to file)
30 100% (source to file)
31 100% (source to file)
32 100% (source to file)
33 100% (source to file)
34 100% (source to file)
35 100% (source to file)
36 100% (source to file)
37 100% (source to file)
38 100% (source to file)
39 100% (source to file)
40 100% (source to file)
41 100% (source to file)
42 100% (source to file)
43 100% (source to file)
44 100% (source to file)
45 100% (source to file)
46 100% (source to file)
47 100% (source to file)
48 100% (source to file)
49 100% (source to file)
50 100% (source to file)
51 100% (source to file)
52 100% (source to file)
53 100% (source to file)
54 100% (source to file)
55 100% (source to file)
56 100% (source to file)
57 100% (source to file)
58 100% (source to file)
59 100% (source to file)
60 100% (source to file)
61 100% (source to file)
62 100% (source to file)
63 100% (source to file)
64 100% (source to file)
65 100% (source to file)
66 100% (source to file)
67 100% (source to file)
68 100% (source to file)
69 100% (source to file)
70 100% (source to file)
71 100% (source to file)
72 100% (source to file)
73 100% (source to file)
74 100% (source to file)
75 100% (source to file)
76 100% (source to file)
77 100% (source to file)
78 100% (source to file)
79 100% (source to file)
80 100% (source to file)
81 100% (source to file)
82 100% (source to file)
83 100% (source to file)
84 100% (source to file)
85 100% (source to file)
86 100% (source to file)
87 100% (source to file)
88 100% (source to file)
89 100% (source to file)
90 100% (source to file)
91 100% (source to file)
92 100% (source to file)
93 100% (source to file)
94 100% (source to file)
95 100% (source to file)
96 100% (source to file)
97 100% (source to file)
98 100% (source to file)
99 100% (source to file)
100 100% (source to file)

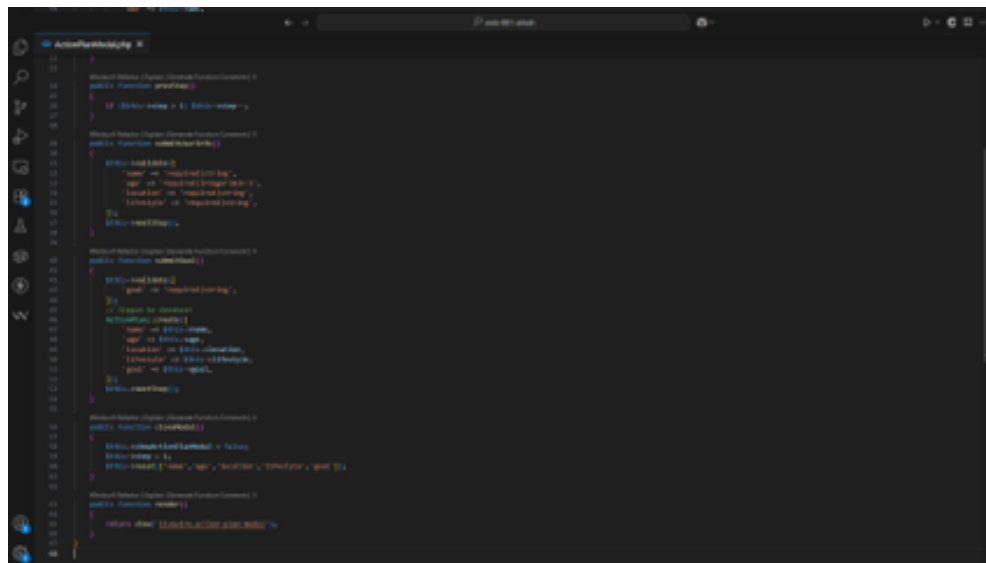
```

[illegible][illegible]

Lampiran 20. Fitur *Action plan*

Fitur yang memungkinkan pengguna membuat rencana aksi untuk mengurangi jejak karbon mereka. Fitur ini diimplementasikan menggunakan komponen Livewire app/Livewire/ActionPlanModal.php. Komponen ini menangani logika untuk *action plan* modal, yang berupa formulir *multi-step* untuk mengumpulkan informasi pengguna seperti nama, usia, lokasi, gaya hidup, dan tujuan. Setelah itu, komponen menyimpan rencana aksi pengguna ke dalam *database*.

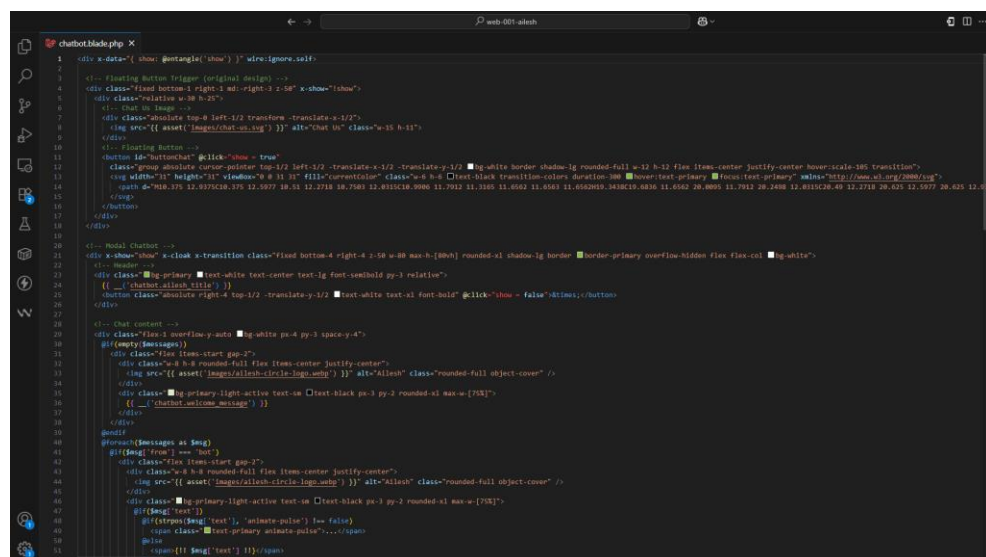




Lampiran 21. Fitur *Chatbot*

Fitur chatbot adalah fitur yang memungkinkan pengguna untuk berinteraksi dengan *chatbot*. Fitur ini diimplementasikan menggunakan komponen Livewire app/Livewire/Chatbot.php. Komponen ini menangani logika *chatbot*, termasuk mengirim dan menerima pesan, serta menyimpan riwayat percakapan ke dalam *database*. Komponen ini juga terintegrasi dengan layanan AI untuk memberikan respons cerdas terhadap pertanyaan pengguna.

- View



```

resources > views > livewire > @chatbot.blade.php
@isset($image) 
41 @if($msg['from'] === 'bot')
42 <div class="flex items-start gap-2">
43 <div class="w-8 h-8 rounded-full flex items-center justify-center">
44 
45 </div>
46 <div class="bg-primary-light-active text-on <div class="px-3 py-2 rounded-xl max-w-[75%]">
47 @if($msg['text'])
48 @if(strpos($msg['text'], 'animate-pulse') !== false)
49 <span class="bg-primary animate-pulse" style="width: 100px; height: 1.2em; display: inline-block;">

```

- Backend

```

1 <?php
2
3 namespace App\Livewire;
4
5 use Livewire\Component;
6 use App\Models\ChatMessage;
7 use Illuminate\Support\Facades\Hash;
8 use Illuminate\Support\Facades\URL;
9 use Illuminate\Support\Facades\Config;
10 use Illuminate\Support\Facades\Log;
11
12 use Livewire\WithFileUploads;
13
14 class Chatbot extends Component
15 {
16     use WithFileUploads;
17
18     public $messages = [];
19     public $newMsg = '';
20     public $sessionId;
21     public $show = false;
22     public $sessionId;
23     public $isLoading = false;
24
25     // Without Refactor (Legacy) (Generate Function Comment) X
26     public function mount()
27     {
28         $this->show = false;
29         $this->sessionId = request()->cookie('chatbot_session_id') ? $this->randomBytes(16) :
30         // Set cookie if not exists
31         if (request()->cookie('chatbot_session_id') == null) {
32             $cookie = response(cookie('chatbot_session_id', $this->randomBytes(16))); // session cookie
33             $this->messages = ChatMessage::where('session_id', $this->sessionId);
34             $cookie->expiresAt = now()->addDays(1);
35             $cookie->secure = true;
36             $cookie->httpOnly = true;
37             $cookie->path = '/';
38         }
39     }
40
41     // Without Refactor (Legacy) (Generate Function Comment) X
42     public function sendMessage()
43     {
44         if (trim($this->newMsg) === '') return;
45         $sessionId = null;
46         if ($this->sessionId) {
47             $sessionId = $this->sessionId;
48             $data = $this->image->getRealPath();
49             $data => base64_encode(file_get_contents($this->image->getRealPath()));
50             $name => $this->image->getClientOriginalName();
51         }
52     }
53 }

```

```

54
55 $msgData = [
56     'from' => 'user',
57     'text' => $this->newMsg,
58     'session_id' => $sessionId,
59 ];
60
61 if ($this->image) {
62     $msgData['image'] = "/storage/" . $this->image->store('chat-images', 'public');
63 }
64
65 $msg = ChatMessage::create($msgData);
66 $this->messages[] = $msg->toArray();
67 $sessionId = $this->sessionId;
68 $this->newMsg = '';
69 // Show loading indicator
70 $this->isLoading = true;
71 $this->messages[] = [
72     'from' => 'bot',
73     'text' => 'user class="animate-pulse" style="width: 100px; height: 1.2em; display: inline-block;">

```

```
Chatbot.php X
109 $session_id => $this->session_id,
110 ];
111 // Inline data hanya jika ada gambar
112 $inline_data = null;
113 if ($this->$image) {
114     $mime_type => $this->$image->getMimeType(),
115     $data => base64_encode(file_get_contents($this->$image->getRealPath()));
116 }
117 $this->botReply['', $inline_data];
118 $this->$image = null;
119 }
120
121 WindowReflector [Export] Generate Function Comment X
122 public function botReply($question, $inline_data = null)
123 {
124     // Rangkai chat terakhir user dan bot (max 3 pesan terakhir)
125     $count = 0;
126     for ($i = count($this->messages) - 1; $i >= 0 && $count < 3; $i--) {
127         $msg = $this->messages[$i];
128         if (empty($msg['text'])) {
129             // Jangan masukkan pesan loading bot ke context
130             if ($msg['from'] === 'bot' && strpos($msg['text'], 'animate-pulse') !== false) {
131                 continue;
132             }
133             // Jangan masukkan pesan 'system sedang sibuk' ke context
134             if (strpos($msg['text']) === 'system sedang sibuk') {
135                 continue;
136             }
137             $count++;
138             $context[] = [
139                 'from' => $msg['from'],
140                 'text' => $msg['text'],
141                 'timestamp' => isset($msg['created_at']) && $msg['created_at'] ? $msg['created_at'] : now()->toDateTimeString();
142             ];
143         }
144     }
145     $context = array_reverse($context); // urutkan dari awal ke akhir
146     $answers = $this->callAIService($question, $inline_data, $context);
147     // Return to client as triple backticks and final 'from bot' answer
148     $cleanhtml = function($text) {
149         if ($text) return $text;
150         // Remove leading/trailing triple backticks and optional 'bot' after first backtick
151         $text = preg_replace("/`(?:`+|\\s)*`/i", "", $text);
152         $text = preg_replace("/`(?:`+|\\s)*`/i", "", $text);
153         return trim($text);
154     };
155     // Remove loading indicator before showing real bot answer
156 }
```

```
Chatbot.php X
148 // Remove loading indicator before showing real bot answer
149 if ($this->$loading) {
150     for ($i = count($this->messages) - 1; $i >= 0; $i--) {
151         if ($this->messages[$i]['from'] === 'bot' && strpos($this->messages[$i]['text'], 'animate-pulse') !== false) {
152             array_splice($this->messages, $i, 1);
153             break;
154         }
155     }
156     $this->$loading = false;
157 }
158 if (is_array($answers)) {
159     // Jika jawaban adalah array berarti jika langsung (bukan array of array)
160     if (array_key_exists('output', $answers)) {
161         $answers = [$answers];
162     }
163     foreach ($answers as $ans) {
164         $msg = ChatMessage::create([
165             'from' => 'bot',
166             'text' => isset($ans['output']) ? $cleanhtml($ans['output']) : (isset($ans['answer']) ? $cleanhtml($ans['answer']) : null),
167             'session_id' => $this->session_id,
168         ]);
169         $this->messages[] = $msg->toArray();
170     }
171 } elseif ($answers === null) {
172     $msg = ChatMessage::create([
173         'from' => 'bot',
174         'text' => 'system sedang sibuk',
175         'session_id' => $this->session_id,
176     ]);
177     $this->messages[] = $msg->toArray();
178 } else {
179     $msg = ChatMessage::create([
180         'from' => 'bot',
181         'text' => $cleanhtml($answers ?? $question),
182         'session_id' => $this->session_id,
183     ]);
184     $this->messages[] = $msg->toArray();
185 }
186
187 WindowReflector [Export] Generate Function Comment X
188 protected function callAIService($question, $inline_data = null)
189 {
190     $url = Config::get('services.ai.service.url');
191     $username = Config::get('services.ai.service.username');
192     $password = Config::get('services.ai.service.password');
193     if (!$url || !$username || !$password) return null;
194     $body = [
195         'question' => $question,
196     ];
197 }
```

```
Chatbot.php X
198 {
199     $url = Config::get('services.ai.service.url');
200     $username = Config::get('services.ai.service.username');
201     $password = Config::get('services.ai.service.password');
202     if (!$url || !$username || !$password) return null;
203     $body = [
204         'question' => $question,
205     ];
206     if ($inline_data) {
207         $body['inline_data'] = $inline_data;
208     }
209     // Kirim context jika ada
210     $args = func_get_args();
211     if (isset($args[2]) && !empty($args[2])) {
212         $body['context'] = $args[2];
213     }
214     $body = ['chatinput' => $body];
215     try {
216         $response = http::withHeaders([
217             'username' => $username,
218             'password' => $password,
219         ])
220         ->acceptJson()
221         ->post($url, $body);
222         if ($response->successful()) {
223             $data = $response->json();
224             return $data;
225         } else {
226             log::error('AI service response not successful', [
227                 'status' => $response->status(),
228                 'body' => $response->body(),
229                 'request' => $body,
230             ]);
231         }
232     } catch (\Exception $e) {
233         log::error('AI service exception', [
234             'error' => $e->getMessage(),
235             'request' => $body,
236         ]);
237     }
238     return null;
239 }
240
241 WindowReflector [Export] Generate Function Comment X
242 public function render()
243 {
244     return view('llm/inline_chatbot');
245 }
246 }
```

Lampiran 22. *Approve Journal*

Fitur *Approver* merupakan menu yang dirancang untuk memfasilitasi proses peninjauan dan persetujuan konten jurnal yang diunggah oleh pengguna. Admin dapat menyetujui ataupun menolak postingan pengguna dengan menyertakan catatan tentang penolakan postingan. Implementasi ini memanfaatkan kerangka kerja Livewire, dengan logika utama diatur dalam kelas PHP Approver.php dan antarmuka pengguna disajikan melalui *template* Blade approver.blade.php.

```

1  <?php
2
3  namespace App\Livewire\Profile;
4
5  use App\Models\Post;
6  use Livewire\Component;
7  use Illuminate\Support\Facades\Auth;
8  use Illuminate\Support\Facades\Storage;
9
10 class Approver extends Component
11 {
12     protected $listeners = ['postAdded' => 'refreshPosts'];
13
14     public $posts;
15     public $postToRejectModel = false;
16     public $showRejectModal = false;
17     public $showApproveModal = false;
18     public $showPostDetailModel = false;
19     public $selectedPost = null;
20     public $page = 1;
21     public $perPage = 5;
22     public $showMore = true;
23     public $showDescriptionModel = false;
24     public $postDescription = '';
25     public $showPostDetailModel = false;
26     public $selectedPost = null;
27
28     public function mount()
29     {
30         $this->posts = collect();
31         $this->loadPosts();
32     }
33
34     public function showDescription($description)
35     {
36         $this->postDescription = $description;
37         $this->showDescriptionModel = true;
38     }
39
40     public function closeDescriptionModel()
41     {
42         $this->showDescriptionModel = false;
43         $this->postDescription = '';
44     }
45
46     public function loadPosts()
47     {
48         $query = Post::with('media')->where('status', 'PMD')->latest();
49         $newPosts = $query->skip(($this->page - 1) * $this->perPage)->take($this->perPage)->get();
50         if ($newPosts->count() < $this->perPage) {
51             $this->showMore = false;
52         }
53         $this->posts = $this->posts->merge($newPosts);
54     }
55
56     public function showPostDetail($postId)
57     {
58         $this->selectedPost = Post::with(['user', 'media'])->find($postId);
59         $this->showPostDetailModel = true;
60     }
61
62     public function closePostDetailModel()
63     {
64         $this->showPostDetailModel = false;
65         $this->selectedPost = null;
66     }
67
68     public function loadMore()
69     {
70         if ($this->showMore) {
71             $this->page++;
72             $this->loadPosts();
73         }
74     }
75
76     public function refreshPosts()
77     {
78         $this->page = 1;
79         $this->posts = collect();
80         $this->showMore = true;
81         $this->loadPosts();
82     }
83
84     public function confirmReject($postId)
85     {
86         $this->postToReject = $postId;
87         $this->showRejectModal = true;
88     }
89
90     public function rejectPost()
91     {
92         $this->validate([
93             'note' => 'required|string|max:255',
94         ]);
95
96         $post = Post::find($this->postToReject);
97
98         if ($post) {
99             $post->update([
100                 'status' => 'RJC',
101                 'reject_note' => $this->note
102             ]);
103         }
104     }
105 }
```

View

[illegible]

```
resources > views > livewire > profile > appviewblade.php
181 </div>
182 <p class="text-on [text-black">
183 @if ($post->getLocales() == 'id')
184 {{ $selectedPost->created_at->locale: 'id' ->translatedFormat('l, d M Y') }} &nbsp;&nbsp;&nbsp; &nbsp;&nbsp;&nbsp;
185 {{ $selectedPost->created_at->locale: 'id' ->format('w.i') }}
186 @else
187 {{ $selectedPost->created_at->locale: 'en' ->format('l, d M Y') }} &nbsp;&nbsp;&nbsp; &nbsp;&nbsp;&nbsp;
188 {{ $selectedPost->created_at->locale: 'en' ->format('w.i.A') }}
189 @endif
190 </p>
191 </div>
192 <div class="flex justify-center mt-2 mb-18">
193 <h1 class="text-2xl md:text-3xl font-sansbold [text-black text-center]">{{ $selectedPost->title }}</h1>
194 </div>
195 <div class="flex justify-center items-center gap-4">
196 @foreach($selectedPost->media as $media)
197 <div class="flex justify-center items-center mb-4">
198 @if($media->type == 'image')
199 
202 @else
203 <video src="{{ asset('storage/' . $media->file_path) }}" controls
204 class="rounded-2xl w-full h-auto object-cover max-w-[150px]" /></video>
205 @endif
206 @endforeach
207 </div>
208 @endif
209 <div class="bg-secondary-light hover:p-4 rounded-xl [text-black leading-relaxed text-on md:text-base max-h-88 md:max-h-158 overflow-y:auto">
210 {{ $selectedPost->description }}
211 </div>
212 <div class="flex justify-center mt-6">
213 <button wire:click="closePostDetailModal"
214 class="bg-[var(--color-secondary)] text-white hover:text-[var(--color-secondary)] hover:bg-secondary-light-active focus:text-[var(--color-secondary)] focus:bg-secondary-light-active">
215 </div>
216 @endif
217 </script>
218 document.addEventListener('DOMContentLoaded', function () {
219 var scrollbar = document.getElementById('journal-scroll');
220 var isLoading = false;
221 if (scrollbar) {
222 scrollbar.addEventListener('scroll', function () {
223 if (
224 scrollbar.scrollTop + scrollbar.clientHeight >= scrollbar.scrollHeight - 10 &&
225 !isLoading
226 ) {
227 isLoading = true;
228 var lw = window.Livewire && window.Livewire.find && window.Livewire.find('{{ $this->getId() }}');
229 if (lw) {
230 lw.call('loadMore');
231 }
232 setTimeout(function () { isLoading = false; }, 700); // debounce agar tidak double call
233 }
234 });
235 </script>
236 </div>
```

Lampiran 23. *Manage One Day, One Tips*

Fitur ini adalah bagian dari pengelolaan konten *One day, One tips* yang berfungsi untuk mengatur dan menampilkan teks serta gambar yang berubah setiap hari, dengan dukungan untuk dua bahasa. Komponen ini dibangun menggunakan kerangka kerja Livewire, di mana logika utamanya diatur dalam DayText.php dan tampilan antarmuka pengguna day-text.blade.php.

```

app > Livewire > Profile > app > DayTest.php > @DayTest > @render()
1 <?php
2
3 namespace App\Livewire\Profile;
4
5 use Livewire\Component;
6 use App\Models\DailyText;
7 use Livewire\WithPagination;
8 use Livewire\WithFileUploads;
9 use Illuminate\Support\Facades\Storage;
10
11 class DayText extends Component
12 {
13     use WithPagination, WithFileUploads;
14
15     public $day, $teks_st_en, $teks_st_id, $teks_rd_en, $teks_rd_id, $teks_rd_en, $teks_rd_id, $teks_th_en, $teks_th_id;
16     public $image_st, $image_md, $image_rd, $image_th;
17     public $dailyTextId;
18     public $isOpen = 0;
19     public $daysOfWeek;
20
21     public function mount()
22     {
23         $this->daysOfWeek = [
24             'monday' => __('daily_text.monday'),
25             'tuesday' => __('daily_text.tuesday'),
26             'wednesday' => __('daily_text.wednesday'),
27             'thursday' => __('daily_text.thursday'),
28             'friday' => __('daily_text.friday'),
29             'saturday' => __('daily_text.saturday'),
30             'sunday' => __('daily_text.sunday'),
31         ];
32     }
33
34     public function render()
35     {
36         $dailyTexts = DailyText::orderBy('created_at')->paginate(10);
37         return view('livewire.profile.day-text', compact('dailyTexts'));
38     }
39
40     public function create()
41     {
42         $this->resetInputFields();
43         $this->openModal();
44     }
45
46     public function openModal()
47     {
48         $this->isOpen = true;
49     }
50
51     public function closeModal()

```

```

app > Livewire > Profile > app > DayTest.php > @DayTest
51
52     public function closeModal()
53     {
54         $this->isOpen = false;
55     }
56
57     private function resetInputFields()
58     {
59         $this->reset([
60             'day',
61             'teks_st_en',
62             'teks_st_id',
63             'teks_rd_en',
64             'teks_rd_id',
65             'teks_th_en',
66             'teks_th_id',
67             'image_st',
68             'image_md',
69             'image_rd',
70             'image_th',
71             'dailyTextId'
72         ]);
73     }
74
75     public function store()
76     {
77         $this->validate([
78             'day' => 'required',
79             'teks_st_en' => 'nullable',
80             'teks_st_id' => 'nullable',
81             'teks_rd_en' => 'nullable',
82             'teks_rd_id' => 'nullable',
83             'teks_th_en' => 'nullable',
84             'teks_th_id' => 'nullable',
85             'image_st' => $this->image_st instanceof \Livewire\Features\SupportFileUploads\TemporaryUploadedFile ? 'nullable:image|max:1024' : 'nullable|string',
86             'image_md' => $this->image_md instanceof \Livewire\Features\SupportFileUploads\TemporaryUploadedFile ? 'nullable:image|max:1024' : 'nullable|string',
87             'image_rd' => $this->image_rd instanceof \Livewire\Features\SupportFileUploads\TemporaryUploadedFile ? 'nullable:image|max:1024' : 'nullable|string',
88             'image_th' => $this->image_th instanceof \Livewire\Features\SupportFileUploads\TemporaryUploadedFile ? 'nullable:image|max:1024' : 'nullable|string',
89         ]);
90
91         $data = [
92             'day' => $this->day,
93             'teks_st_en' => $this->teks_st_en,
94             'teks_st_id' => $this->teks_st_id,
95             'teks_rd_en' => $this->teks_rd_en,
96             'teks_rd_id' => $this->teks_rd_id,
97             'teks_th_en' => $this->teks_th_en,
98             'teks_th_id' => $this->teks_th_id,
99             'image_st' => $this->image_st,
100             'image_md' => $this->image_md,
101             'image_rd' => $this->image_rd,
102             'image_th' => $this->image_th

```

```

app > Livewire > Profile > app > DayTest.php > @DayTest > @store()
101     'teks_th_en' => $this->teks_th_en,
102     'teks_th_id' => $this->teks_th_id,
103 ];
104
105 if ($this->image_st && is_string($this->image_st)) {
106     $data['image_st'] = $this->image_st->store('daily_texts', 'public');
107 }
108 if ($this->image_md && is_string($this->image_md)) {
109     $data['image_md'] = $this->image_md->store('daily_texts', 'public');
110 }
111 if ($this->image_rd && is_string($this->image_rd)) {
112     $data['image_rd'] = $this->image_rd->store('daily_texts', 'public');
113 }
114 if ($this->image_th && is_string($this->image_th)) {
115     $data['image_th'] = $this->image_th->store('daily_texts', 'public');
116 }
117
118 DailyText::updateOrCreate(['id' => $this->dailyTextId], $data);
119
120 session()->flash('message',
121     $this->dailyTextId ? __('daily_text.updated_success') : __('daily_text.created_success');
122 );
123 $this->closeModal();
124 $this->resetInputFields();
125
126 public function edit($id)
127 {
128     $dailyText = DailyText::findOrFail($id);
129     $this->dailyTextId = $id;
130     $this->day = $dailyText->day;
131     $this->teks_st_en = $dailyText->teks_st_en;
132     $this->teks_st_id = $dailyText->teks_st_id;
133     $this->teks_rd_en = $dailyText->teks_rd_en;
134     $this->teks_rd_id = $dailyText->teks_rd_id;
135     $this->teks_th_en = $dailyText->teks_th_en;
136     $this->teks_th_id = $dailyText->teks_th_id;
137     $this->image_st = $dailyText->image_st;
138     $this->image_md = $dailyText->image_md;
139     $this->image_rd = $dailyText->image_rd;
140     $this->image_th = $dailyText->image_th;
141
142     $this->openModal();
143 }
144
145 public function delete($id)
146 {
147     $dailyText = DailyText::findOrFail($id);
148     if ($dailyText->image_st->isNotEmpty()) {
149         $dailyText->image_st->delete();
150     }

```

View

View

Lampiran 24. Survei UEQ Melalui *Google Form*



Survei Pengalaman Pengguna Terhadap Desain Antarmuka Website Ailesh

Annyeong 🙋

Formulir ini disusun sebagai bagian dari penelitian skripsi saya, Ivana Theophilie dengan tujuan untuk mengukur pengalaman pengguna terhadap desain antarmuka website Ailesh menggunakan metode User Experience Questionnaire (UEQ).

Survei ini mencakup aspek-aspek seperti kemudahan penggunaan, kenyamanan, daya tarik visual, serta kepuasan pengguna secara keseluruhan.

Data yang diperoleh dari survei ini akan digunakan untuk menganalisis kualitas desain antarmuka website Ailesh dan menjadi dasar dalam memberikan rekomendasi perbaikan yang lebih tepat sasaran.

Responden diharapkan menjawab pertanyaan berdasarkan pengalaman langsung saat mengakses website Ailesh: <https://devsec.donnvdlat.my.id/>

Seluruh data yang dikumpulkan akan dijaga kerahasiaannya dan hanya digunakan untuk kepentingan penelitian akademik.

Terima kasih atas partisipasinya ❤️

indonesianasa74@gmail.com [Switch account](#)

* Indicates required question

Pendidikan terakhir *

☐ SD/MI

☐ SMP/MTS

☐ SMA/MAN/SMK/Kursus

☐ Sarjana/S1

☐ Magister/S2

☐ Doktor/S3

☐ Other: _____

Pekerjaan *

Your answer _____

Perangkat yang digunakan dalam mengakses website *

☐ Smartphone/HP

☐ Laptop/Komputer

Next **Clear form**

Email *

Your email _____

Nama *

Your answer _____

Usia *

Your answer _____

Jenis Kelamin *

☐ Laki-laki

☐ Perempuan



Survei Pengalaman Pengguna Terhadap Desain Antarmuka Website Ailesh

indonesianasa74@gmail.com [Switch account](#)

* Indicates required question

Silahkan jawab pertanyaan yang tersedia sesuai pengalaman Anda setelah menjelajah website Ailesh

Aturan pengisian survei :

- Bacalah setiap pasangan kata dengan teliti dan pilih nilai yang paling sesuai dengan pendapat Anda.
- Skala penilaian berkisar dari -3 (sisi kiri/sangat tidak setuju), 0 (sisi tengah/netral), hingga +3 (sisi kanan/sangat setuju).
- Survei ini bersifat anonim dan data hanya digunakan untuk keperluan penelitian.
- Mohon isi seluruh bagian survei tanpa dikosongkan.

Partisipasi Anda sangat membantu dalam pengembangan website Ailesh yang lebih baik. **Enjoy your task** 😊

Pilih jawaban pada skala yang paling sesuai dengan pendapat Anda

Good luck 😊

Apakah penggunaan website terasa merepotkan (menyusahkan) atau justru menyenangkan? *

1 2 3 4 5 6 7

Menyusahkan ☐ ☐ ☐ ☐ ☐ ☐ ☐ Menyenangkan

Apakah isi dan fungsi website mudah dimengerti (dipahami)? *

1 2 3 4 5 6 7

Tidak dapat dipahami ☐ ☐ ☐ ☐ ☐ ☐ ☐ Dapat dipahami

Apakah tampilan dan kontennya membosankan (monoton) atau bervariasi dan menarik (kreatif)? *

1 2 3 4 5 6 7

Monoton ☐ ☐ ☐ ☐ ☐ ☐ ☐ Kreatif

Seberapa cepat Anda memahami (mempelajari) cara menggunakan website ini? *

1 2 3 4 5 6 7

Sulit dipelajari ☐ ☐ ☐ ☐ ☐ ☐ ☐ Mudah dipelajari

Apakah informasi yang diberikan berguna (bermanfaat) bagi Anda? *

1 2 3 4 5 6 7

Kurang bermanfaat ☐ ☐ ☐ ☐ ☐ ☐ ☐ Bermanfaat

Apakah Anda merasa tertarik dan betah menjelajah website ini? *

1 2 3 4 5 6 7

Membosankan ☐ ☐ ☐ ☐ ☐ ☐ ☐ Mengasyikkan

Apakah tampilan dan isi website membuat Anda tertarik? *

1 2 3 4 5 6 7

Tidak menarik ☐ ☐ ☐ ☐ ☐ ☐ ☐ Menarik

Apakah hasil dari tindakan Anda di website mudah ditebak dan konsisten? *

(Saat pengguna mengklik tombol, memilih menu, atau mengisi formulir, apakah hasil atau respon dari website sesuai harapan dan tidak membingungkan)

1 2 3 4 5 6 7

Tidak dapat diprediksi ☐ ☐ ☐ ☐ ☐ ☐ ☐ Dapat diprediksi

Apakah waktu loading atau respon website terasa cepat? *

1 2 3 4 5 6 7

Lambat ☐ ☐ ☐ ☐ ☐ ☐ ☐ Cepat

Apakah tampilan website terlihat kreatif atau terlalu standar? *

(Berdaya cipta = unik dan penuh ide baru, konvensional = biasa dan umum)

1 2 3 4 5 6 7

Berdaya cipta ☐ ☐ ☐ ☐ ☐ ☐ ☐ Konvensional

Apakah desain website membantu Anda dalam menggunakan fitur-fitur yang ada? *

1 2 3 4 5 6 7

Menghalangi ☐ ☐ ☐ ☐ ☐ ☐ ☐ Mendukung

Secara keseluruhan, apakah kualitas website baik atau buruk? *

1 2 3 4 5 6 7

Buruk ☐ ☐ ☐ ☐ ☐ ☐ ☐ Baik

Apakah website mudah digunakan tanpa membingungkan? *

1 2 3 4 5 6 7

Rumit ☐ ☐ ☐ ☐ ☐ ☐ ☐ Sederhana

Back Next Clear form

Tidak atraktif ○ ○ ○ ○ ○ ○ ○ Atraktif

[illegible]



SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER

AMIKOM SURAKARTA

www.amikomsolo.ac.id amikomsolo@amikomsolo.ac.id

LEMBAR REVISI

UJIAN SKRIPSI

TA. 2024/2025

NAMA : IVANA THEOPHILIA ONI DAWISTRI
NOMOR MAHASISWA : 2113010284
PROGRAM STUDI : Informatika
HARI, TGL UJIAN : Selasa, 02 September 2025
BATAS AKHIR REVISI : **Selasa, 16 September 2025**
JUDUL SKRIPSI : Pengembangan Desain Antarmuka Pengguna Website
Ailesh Menggunakan Metode Design Thinking
PEMBIMBING SKRIPSI : 1. Ina Sholihah Widiati, M.Kom.
2. Lilik Sugiarto, M.Kom

NO	URAIAN	KET
1	Sesuaikan penulisan dengan template	l ₁
2	Berikan penyelesaian pada setiap gambar/ tabel	l ₁
3	Sampaikan hasil evaluasi repositori	l ₁
4	Sampaikan hasil evaluasi pengujian secara lengkap	l ₁
5	Sampaikan hasil pengumpulan data pada lampiran	l ₁

Penguji

: Ryhan Abdul Aziz

(l₁ , *)

*) Ditandatangani oleh penguji setelah selesai Ujian Skripsi

Telah selesai dilakukan revisi pada tanggal : 12 September 2025

Penguji

: Ryhan Abdul Aziz

(l₁ , **)

**) Ditandatangani oleh penguji setelah mahasiswa selesai melakukan bimbingan revisi ke penguji
Jika revisi dilakukan setelah batas waktu revisi, maka akan diujikan ulang



SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER

AMIKOM SURAKARTA

www.amikomsolo.ac.id

amikomsolo@amikomsolo.ac.id

**LEMBAR REVISI
UJIAN SKRIPSI**

TA. 2024/2025

NAMA : IVANA THEOPHILIA ONI DAWISTRI
NOMOR MAHASISWA : 2113010284
PROGRAM STUDI : Informatika
HARI, TGL UJIAN : Selasa, 02 September 2025
BATAS AKHIR REVISI : Selasa, 16 September 2025
JUDUL SKRIPSI : Pengembangan Desain Antarmuka Pengguna Website
Ailesh Menggunakan Metode Design Thinking
PEMBIMBING SKRIPSI : 1. Ina Sholihah Widiati, M.Kom.
2. Lilik Sugiarto, M.Kom

NO	URAIAN	KET
	LBM belum jelas	2
	Tinjauan Pustaka & landasan Teori	2
	Metode Penelitian Pengumpulan data (Point)	2
	Bab IV cek dan sesuaikan per tahapan Gambar Standar Eua kasih di bab 3 / landasan (Pengumpulan data) dataset	2
	Program : Typografi → Tulisan perlu diperhatikan database	2

Penguji : Indrawan Ady S., M.Kom () *

) Ditandatangani oleh penguji setelah selesai Ujian Skripsi

Telah selesai dilakukan revisi pada tanggal : 12 September 2025

Penguji : Indrawan Ady S., M.Kom () **)

**) Ditandatangani oleh penguji setelah mahasiswa selesai melakukan bimbingan
revisi ke penguji
Jika revisi dilakukan setelah batas waktu revisi, maka akan diujikan ulang



SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER

AMIKOM SURAKARTA

www.amikomsolo.ac.id

amikomsolo@amikomsolo.ac.id

**LEMBAR REVISI
UJIAN SKRIPSI**

TA. 2024/2025

NAMA : IVANA THEOPHILIA ONI DAWISTRI
NOMOR MAHASISWA : 2113010284
PROGRAM STUDI : Informatika
HARI, TGL UJIAN : Selasa, 02 September 2025
BATAS AKHIR REVISI : **Selasa, 16 September 2025**
JUDUL SKRIPSI : Pengembangan Desain Antarmuka Pengguna Website
Ailesh Menggunakan Metode Design Thinking
PEMBIMBING SKRIPSI : 1. Ina Sholihah Widiati, M.Kom.
2. Lilik Sugiarto, M.Kom

NO	URAIAN	KET
1	ke penguji.	

Penguji : LILIK SUGIARTO, M.Kom () *)

*) Ditandatangani oleh penguji setelah selesai Ujian Skripsi

Telah selesai dilakukan revisi pada tanggal : 12 September 2025

Penguji : Lilik S () **)

**) Ditandatangani oleh penguji setelah mahasiswa selesai melakukan bimbingan revisi ke penguji
Jika revisi dilakukan setelah batas waktu revisi, maka akan diujikan ulang