

SECURITY

SPIS TREŚCI

Spis treści	1
Cel zajęć.....	1
Rozpoczęcie	1
Uwaga	1
Prezentacja tworzenia CRUD.....	Błąd! Nie zdefiniowano zakładki.
Budowa CRUD	Błąd! Nie zdefiniowano zakładki.
Lokalizacja	Błąd! Nie zdefiniowano zakładki.
Pomiary	Błąd! Nie zdefiniowano zakładki.
Commit projektu do GIT.....	20
Podsumowanie.....	21

CEL ZAJĘĆ

Celem głównym zajęć jest zdobycie umiejętności implementacji i konfiguracji logowania i wylogowywania użytkowników oraz ograniczania dostępu do akcji.

ROZPOCZĘCIE

Rozpoczęcie zajęć. Powtórzenie security is_granted w Twig i atrybutach kontrolerów.

Wejściówka?

UWAGA

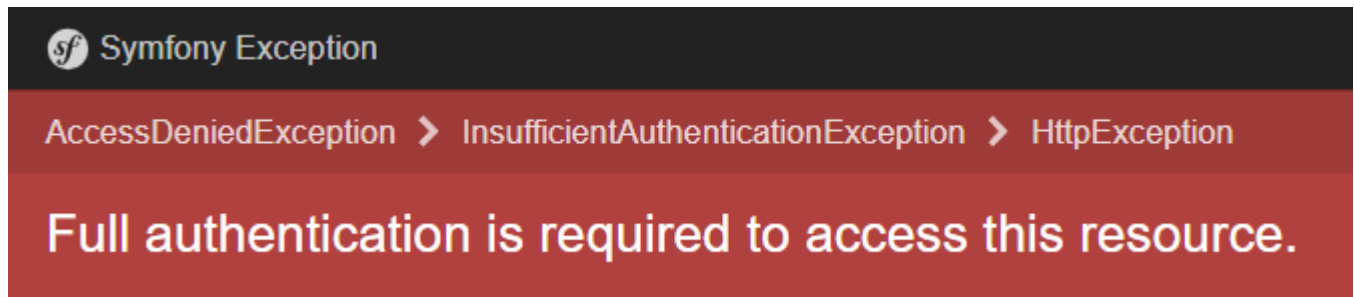
Ten dokument aktywnie wykorzystuje niestandardowe właściwości. Podobnie jak w LAB A wejdź do **Plik** -> **Informacje** -> **Właściwości** -> **Właściwości zaawansowane** -> **Niestandardowe** i zaktualizuj pola. Następnie uruchom ten dokument ponownie lub **Ctrl+A** -> **F9**.

ZABEZPIECZENIE DOSTĘPU DO AKCJI I WIDOKÓW

Obecnie dostęp do wszystkich akcji w systemie jest publicznie otwarty. W tej sekcji zabezpieczymy dostęp. Otwórz plik `src/Controller/LocationController.php` i edytuj akcję `new`:

```
#[Route('/new', name: 'app_location_new', methods: ['GET', 'POST'])]
#[IsGranted('ROLE_LOCATION_NEW')]
public function new(Request $request, EntityManagerInterface $entityManager): Response
{
    $location = new Location();
    $form = $this->createForm(LocationType::class, $location, [
        'validation_groups' => 'create',
    ]);
    $form->handleRequest($request);
}
```

Odwiedź teraz w przeglądarce stronę <http://pogodynka.localhost:00000/location>. Powinna wyświetlić się lista lokalizacji. Następnie kliknij w przycisk tworzenia nowej lokalizacji. Powinien wyświetlić się błąd:



Dlaczego lista lokalizacji wciąż działa, a akcja tworzenia lokalizacji wyrzuca wyjątek?

Dlatego że zablokowaliśmy tylko route do /new, teraz trzeba mieć role ROLE_LOCATION_NEW, aby być w stanie otworzyć akcje tworzenia lokalizacji.

Teraz zmodyfikuj widok listy lokalizacji w pliku templates/location/index.html.twig:

```
--- a/templates/location/index.html.twig
+++ b/templates/location/index.html.twig
@@ -38,5 +38,7 @@
     </tbody>
 </table>

-    <a href="{{ path('app_location_new') }}">Create new</a>
+    {% if is_granted('ROLE_LOCATION_NEW') %}
+    <a href="{{ path('app_location_new') }}">Create new</a>
+    {% endif %}
+{% endblock %}
```

Ponownie wejdź w przeglądarce na listę lokalizacji. Przycisk tworzenia nowej lokalizacji powinien teraz zniknąć.

Dodaj wymuszanie posiadania przez użytkownika roli dla wszystkich akcji kontrolerów lokalizacji i pomiarów. Zabronione jest korzystanie z roli typu ROLE_ADMIN. Zamiast tego utwórz dla każdej akcji w każdym kontrolerze osobną rolę odnoszącą się do czynności, w postaci ROLE_<kontroler>_<akcja>, np. ROLE_LOCATION_INDEX.

Poniżej wstaw zrzuty ekranu kodu atrybutów i sygnatury każdej akcji każdego kontrolera:

```
LocationController::new()
```

```
#[Route('/new', name: 'app_location_new', methods: ['GET', 'POST'])]
#[IsGranted('ROLE_LOCATION_NEW')]
public function new(Request $request, EntityManagerInterface $entityManager): Response
```

LocationController::index():

```
#[Route('/', name: 'app_location_index', methods: ['GET'])]
#[IsGranted('ROLE_LOCATION_INDEX')]
public function index(LocationRepository $locationRepository): Response
{
```

LocationController::show():

```
#[Route("/{id}", name: 'app_location_show', methods: ['GET'])]
#[IsGranted('ROLE_LOCATION_SHOW')]
public function show(Location $location): Response
{
```

LocationController::edit():

```
#[Route("/{id}/edit", name: 'app_location_edit', methods: ['GET', 'POST'])]
#[IsGranted('ROLE_LOCATION_EDIT')]
public function edit(Request $request, Location $location, EntityManagerInterface $entityManager): Response
{
```

LocationController::delete():

```
#[Route("/{id}", name: 'app_location_delete', methods: ['POST'])]
#[IsGranted('ROLE_LOCATION_DELETE')]
public function delete(Request $request, Location $location, EntityManagerInterface $entityManager): Response
```

MeasurementController::new():

```
#[Route('/new', name: 'app_measurement_new', methods: ['GET', 'POST'])]
#[IsGranted('ROLE_MEASUREMENT_NEW')]
public function new(Request $request, EntityManagerInterface $entityManager): Response
{
```

MeasurementController::index():

```
#[Route('/', name: 'app_measurement_index', methods: ['GET'])]
#[IsGranted('ROLE_MEASUREMENT_INDEX')]
public function index(MeasurementRepository $measurementRepository): Response
{
```

MeasurementController::show():

```
#[Route('/{id}', name: 'app_measurement_show', methods: ['GET'])]
#[IsGranted('ROLE_MEASUREMENT_SHOW')]
public function show(Measurement $measurement): Response
```

MeasurementController::edit():

```
#[Route('/{id}/edit', name: 'app_measurement_edit', methods: ['GET', 'POST'])]
#[IsGranted('ROLE_MEASUREMENT_EDIT')]
public function edit(Request $request, Measurement $measurement, EntityManagerInterface $entityManager): Response
{
```

MeasurementController::delete():

```
#[Route('/{id}', name: 'app_measurement_delete', methods: ['POST'])]
#[IsGranted('ROLE_MEASUREMENT_DELETE')]
public function delete(Request $request, Measurement $measurement, EntityManagerInterface $entityManager): Response
{
```

Punkty:	0	1
---------	---	---

Dodaj weryfikację ról w szablonach TWIG na liście lokalizacji i liście pomiarów.

Wstaw zrzut ekranu kodu pliku `templates/location/index.html.twig`:

```

1  {% extends 'base.html.twig' %}
2
3  {% block title %}Location index{% endblock %}
4
5  {% block body %}
6      <div class="container mt-4">
7          <h1 class="display-4">Location Index</h1>
8          <table class="table table-striped">
9              <thead>
10                 <tr>
11                     <th>Id</th>
12                     <th>City</th>
13                     <th>Country</th>
14                     <th>Latitude</th>
15                     <th>Logitude</th>
16                     <th>actions</th>
17                 </tr>
18             </thead>
19             <tbody>
20                 {% for location in locations %}
21                     <tr>
22                         <td>{{ location.id }}</td>
23                         <td>{{ location.city }}</td>
24                         <td>{{ location.country }}</td>
25                         <td>{{ location.latitude }}</td>
26                         <td>{{ location.logitude }}</td>
27                         <td>
28                             {% if is_granted('ROLE_LOCATION_SHOW') %}
29                             <a class="btn btn-primary" href="{{ path('app_location_show', {'id': location.id}) }}">show</a>
30                             {% endif %}
31                             {%if is_granted('ROLE_LOCATION_EDIT') %}
32                             <a class="btn btn-warning" href="{{ path('app_location_edit', {'id': location.id}) }}">edit</a>
33                             {% endif %}
34                         </td>
35                     </tr>
36                 {% else %}
37                     <tr>
38                         <td colspan="6">no records found</td>
39                     </tr>
40                 {% endfor %}
41             </tbody>
42         </table>
43         {% if is_granted('ROLE_LOCATION_NEW ') %}
44             <a class="btn btn-success" href="{{ path('app_location_new') }}">Create new</a>
45         {% endif %}
46     </div>
47 {% endblock %}

```

Wstaw zrzut ekranu kodu pliku `templates/measurement/index.html.twig`:

```

1  You, 1 sekunde temu | 1 author (You)
2  {% extends 'base.html.twig' %}
3  {% block title %}Measurement index{% endblock %}
4
5  {% block body %}
6      <div class="container mt-4">
7          <h1 class="display-4">Measurement index</h1>
8
9          <table class="table table-striped">
10             <thead>
11                 <tr>
12                     <th>Id</th>
13                     <th>Date</th>
14                     <th>Celsius</th>
15                     <th>actions</th>
16                 </tr>
17             </thead>
18             <tbody>
19                 {% for measurement in measurements %}
20                     <tr>
21                         <td>{{ measurement.id }}</td>
22                         <td>{{ measurement.date ? measurement.date|date('Y-m-d') : '' }}</td>
23                         <td>{{ measurement.celsius }}</td>
24                         <td>
25                             {% if is_granted('ROLE_MEASUREMENT_SHOW') %}
26                             <a class="btn btn-primary" href="{{ path('app_measurement_show', {'id': measurement.id}) }}">show</a>
27                             {% endif %}
28                             {% if is_granted('ROLE_MEASUREMENT_EDIT') %}
29                             <a class="btn btn-warning" href="{{ path('app_measurement_edit', {'id': measurement.id}) }}">edit</a>
30                             {% endif %}
31                         </td>
32                     </tr>
33                 {% else %}
34                     <tr>
35                         <td colspan="4">no records found</td>
36                     </tr>
37                 {% endfor %}
38             </tbody>
39         </table>
40         {% if is_granted('ROLE_MEASUREMENT_NEW') %}
41             <a class="btn btn-success" href="{{ path('app_measurement_new') }}">Create new</a>
42         {% endif %}
43     </div>
44     You, 4 tygodnie temu • labD
45 {% endblock %}

```

Punkty:

0

1

PROSTE LOGOWANIE BASIC AUTH

W tej sekcji dodamy możliwość logowania użytkownika poprzez Basic Auth (login i hasło podawane bezpośrednio w adresie URL – `http://login:pass@pogodynka.localhost:00000` – lub w wyświetlonym przez przeglądarkę okienku).

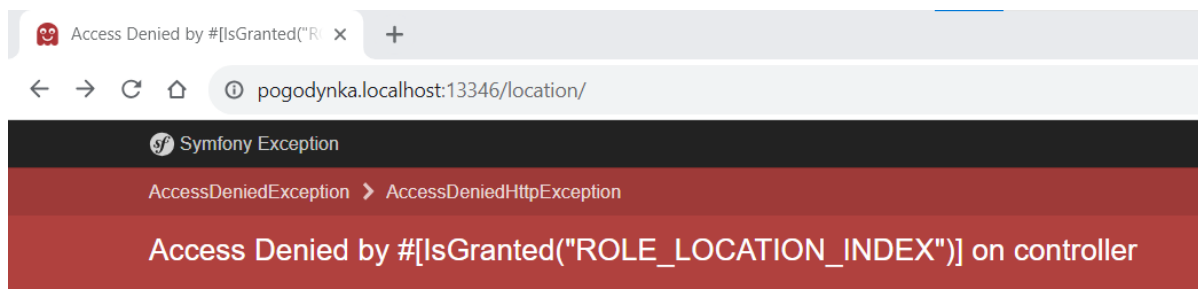
Modyfikuj plik `config/packages/security.yaml`:

AI2 LAB E – Nowakowski Michał – Wersja 1

```
--- a/config/packages/security.yaml
+++ b/config/packages/security.yaml
@@ -1,10 +1,17 @@
 security:
     # https://symfony.com/doc/current/security.html#registering-the-user-hashing-passwords
     password_hashers:
-        Symfony\Component\Security\Core\User\PasswordAuthenticatedUserInterface: 'auto'
+        Symfony\Component\Security\Core\User\PasswordAuthenticatedUserInterface: 'plaintext'
     # https://symfony.com/doc/current/security.html#loading-the-user-the-user-provider
     providers:
-        users_in_memory: { memory: null }
+        users_in_memory:
+            memory:
+                users:
+                    admin:
+                        password: '12345678'
+                        roles:
+                            - 'ROLE_ADMIN'
+                            - 'ROLE_USER'
     firewalls:
         dev:
             pattern: ^/(_(profiler|wdt)|css|images|js)/
@@ -12,6 +19,8 @@ security:
     main:
         lazy: true
         provider: users_in_memory
+        http_basic:
+            realm: 'My Secured Area'

     # activate different ways to authenticate
     # https://symfony.com/doc/current/security.html#the-firewall
@@ -22,8 +31,7 @@ security:
     # Easy way to control access for large sections of your site
     # Note: Only the *first* access control that matches will be used
     access_control:
-        # - { path: ^/admin, roles: ROLE_ADMIN }
-        # - { path: ^/profile, roles: ROLE_USER }
+        - { path: ^/, roles: PUBLIC_ACCESS }
```

W efekcie po wejściu na stronę powinno wyświetlić się okienko logowania Basic Auth. Po poprawnym logowaniu wyświetlony zostanie jednak błąd braku dostępu:



Wynika to z faktu, że obecnie użytkownik admin ma tylko role ROLE_ADMIN i ROLE_USER, a wymagana jest rola ROLE_LOCATION_INDEX. W dalszej części tego laboratorium rozwiążemy ten problem. Na ten moment można tymczasowo zmienić wymaganą rolę w LocationController::index() na ROLE_ADMIN, w celu weryfikacji poprawności działania:

Location index

Id	City	Country	Latitude	Longitude	actions
1	Szczecin	PL	53.4289	14.553	weather
2	Police	PL	53.5521	14.5718	weather

200 @app_location_i 3830 ms 50.0 MiB 1 2 admin 20 ms 3 6.3.4

Zwróć uwagę, że chociaż lista się pojawiła, wszystkie linki zabezpieczone rolami `ROLE_<kontroler>_<akcja>` są ukryte.

Wstaw zrzut ekranu wyskakującego okienka Basic Auth na tle Twojej strony:

localhost:49353

Ta witryna prosi o zalogowanie się.

Nazwa użytkownika

Hasło

Zaloguj się Anuluj

Punkty:	0	1
---------	---	---

W dalszej części wykorzystamy komendę `security:hash-password` do zaszyfrowania hasła przed umieszczeniem go w pliku konfiguracyjnym:

```
php .\bin\console security:hash-password

Symfony Password Hash Utility
=====

Type in your password to be hashed:
>

! [NOTE] The command will take care of generating a salt for you. Be aware that some hashers advise to let them
        generate their own salt. If you're using one of those hashers, please answer 'no' to the question below.
!       Provide the 'empty-salt' option in order to let the hasher handle the generation itself.
```


AI2 LAB E – Nowakowski Michał – Wersja 1

```
Confirm salt generation ? (yes/no) [yes]:  
> no
```

Key	Value
Hasher used	Symfony\Component\PasswordHasher\Hasher\PlaintextPasswordHasher

```
[OK] Password hashing succeeded
```

Zwróć uwagę, że zaszyfrowane hasło to wciąż plaintext. Wynika to z konfiguracji w pliku `security.yaml`. Zmień w nim opcję `plaintext` z powrotem na `auto` i ponów hashowanie. Wygenerowany hash podstaw jako hasło użytkownika w pliku `security.yaml`.

Wstaw zrzut ekranu konfiguracji `security.yaml` z hasłem w postaci uzyskanego hashu:

```
! security.yaml M x  
pogodynka > config > packages > ! security.yaml  
You, 13 sekund temu | 1 author (You)  
1 security:  
2 # https://symfony.com/doc/current/security.html#registering-the-user-hashing-passwords  
3 password_hashers:  
4     Symfony\Component\Security\Core\User\PasswordAuthenticatedUserInterface: 'auto'  
5 # https://symfony.com/doc/current/security.html#loading-the-user-the-user-provider  
6 providers:  
7     users_in_memory:  
8         memory:  
9             users:  
10                admin:  
11                    password: '$2y$13$uQ5f1PBbsbhIdaKdqsZkz.E.YeFWZK8Wyx5kyjPvxjZAXbEQcLL2i' #12345678  
12                    roles:  
13                        - 'ROLE_ADMIN'  
14                        - 'ROLE_USER'  
15                        - 'ROLE_LOCATION_NEW'  
16                        - 'ROLE_LOCATION_EDIT'  
17                        - 'ROLE_LOCATION_DELETE'  
18                        - 'ROLE_LOCATION_INDEX'  
19                        - 'ROLE_LOCATION_SHOW'  
20                user:  
21                    password: '$2y$13$uQ5f1PBbsbhIdaKdqsZkz.E.YeFWZK8Wyx5kyjPvxjZAXbEQcLL2i' #12345678  
22                    roles:  
23                        - 'ROLE_USER'  
24                        - 'ROLE_LOCATION_INDEX'  
25                        - 'ROLE_LOCATION_SHOW'  
26                user2:  
27                    password: '$2y$13$uQ5f1PBbsbhIdaKdqsZkz.E.YeFWZK8Wyx5kyjPvxjZAXbEQcLL2i' #12345678  
28                    roles:  
29                        - 'ROLE_USER'  
30                        - 'ROLE_LOCATION_INDEX'  
31  
32 firewalls:  
33     dev:  
34         pattern: ^/(_(profiler|wdt)|css|images|js)/  
35         security: false  
You, 1 sekundę temu • Uncommitted changes  
DANE WYJŚCIOWE KONSOLA DEBUGOWANIA TERMINAL PORTY GITLENS PROBLEMY  
powershell - pogodynka
```

Punkty:

0

1

UŻYTKOWNICY BAZODANOWI

W tej sekcji usuniemy użytkownika z `security.yaml` na rzecz użytkowników przechowywanych w bazie danych.

Wykonaj komendę `make:user`:

```
php .\bin\console make:user

The name of the security user class (e.g. User) [User]:
>

Do you want to store user data in the database (via Doctrine)? (yes/no) [yes]:
>

Enter a property name that will be the unique "display" name for the user (e.g. email, username, uuid) [email]:
> username

Will this app need to hash/check user passwords? Choose No if passwords are not needed or will be checked/hashed
by some other system (e.g. a single sign-on server).

Does this app need to hash/check user passwords? (yes/no) [yes]:
>

created: src/Entity/User.php
created: src/Repository/UserRepository.php
updated: src/Entity/User.php
updated: config/packages/security.yaml

Success!

Next Steps:
- Review your new App\Entity\User class.
- Use make:entity to add more fields to your User entity and then run make:migration.
- Create a way to authenticate! See https://symfony.com/doc/current/security.html
```

Zaktualizuj schemat bazy danych:

```
php bin\console doctrine:schema:update --dump-sql --force
```

Dodaj co najmniej jednego użytkownika bezpośrednio w bazie danych:

	id	username	roles	password
1	1	admin	["ROLE_ADMIN"]	\$2y\$13\$fhE1St0yxs0GnGc.ygWmt.c1A7...

Zaloguj się poprzez Basic Auth na użytkownika, którego dane zapisane są w bazie danych. W pasku profilera kliknij na nazwę użytkownika. Otworzy się panel Security.

Wstaw zrzut ekranu panelu Security z danymi zalogowanego użytkownika:

Punkty:

0

1

FORMULARZ LOGOWANIA

W tej sekcji wykorzystamy wbudowany w Symfony mechanizm `form_login` do obsługi formularza logowania. Wykorzystaj komendę `make:controller` do utworzenia kontrolera logowania:

```
php .\bin\console make:controller Login
created: src/Controller/LoginController.php
created: templates/login/index.html.twig
```

Success!

Next: Open your new controller class and add some pages!

Zaktualizuj ustawienia firewalla `main` w `security.yaml`. Zastąp logowanie `http_basic` przez `form_login`:

<code>main:</code>	16	16	<code>main:</code>
<code> lazy: true</code>	17	17	<code> lazy: true</code>
<code> provider: app_user_provider</code>	18	18	<code> provider: app_user_provider</code>
<code> http_basic:</code>	19	19	<code> form_login:</code>
<code> realm: 'My Secured Area'</code>	20	20	<code> login_path: app_login</code>
	21	21	<code> check_path: app_login</code>

Teraz zmodyfikuj `LoginController`:

```

class LoginController extends AbstractController
{
    #[Route('/login', name: 'app_login')]
    public function index(AuthenticationUtils $authenticationUtils): Response
    {
        // get the login error if there is one
        $error = $authenticationUtils->getLastAuthenticationError();

        // last username entered by the user
        $lastUsername = $authenticationUtils->getLastUsername();

        return $this->render('login/index.html.twig', [
            'controller_name' => 'LoginController',
            'last_username' => $lastUsername,
            'error'           => $error,
        ]);
    }
}

```

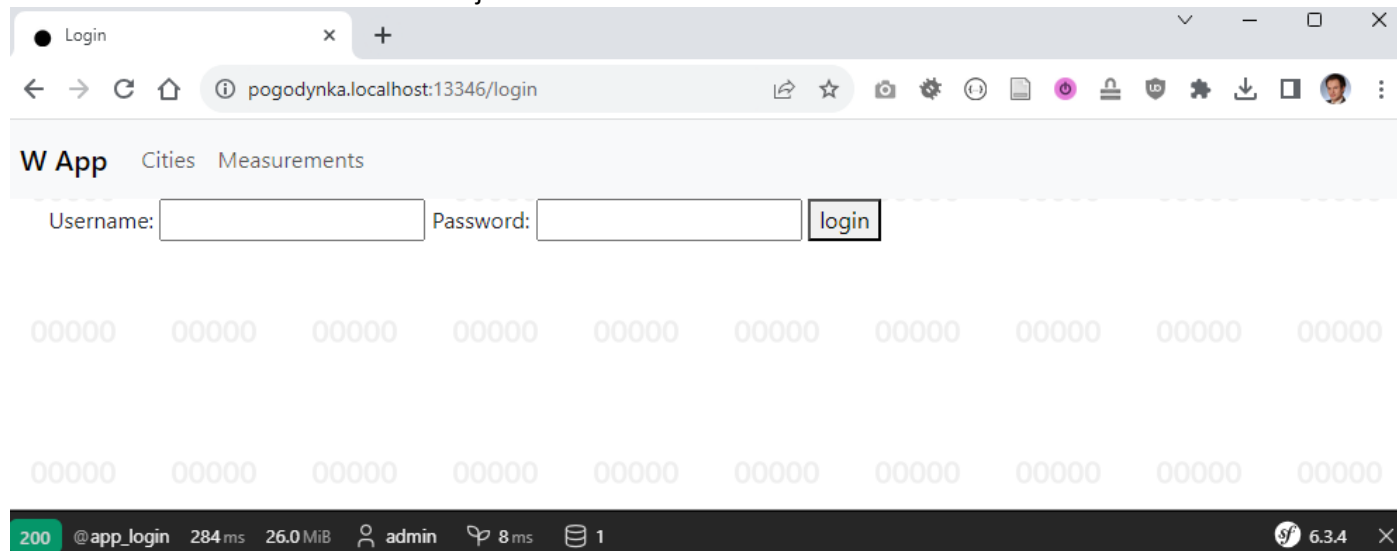
Zmodyfikuj również szablon akcji logowania:

```

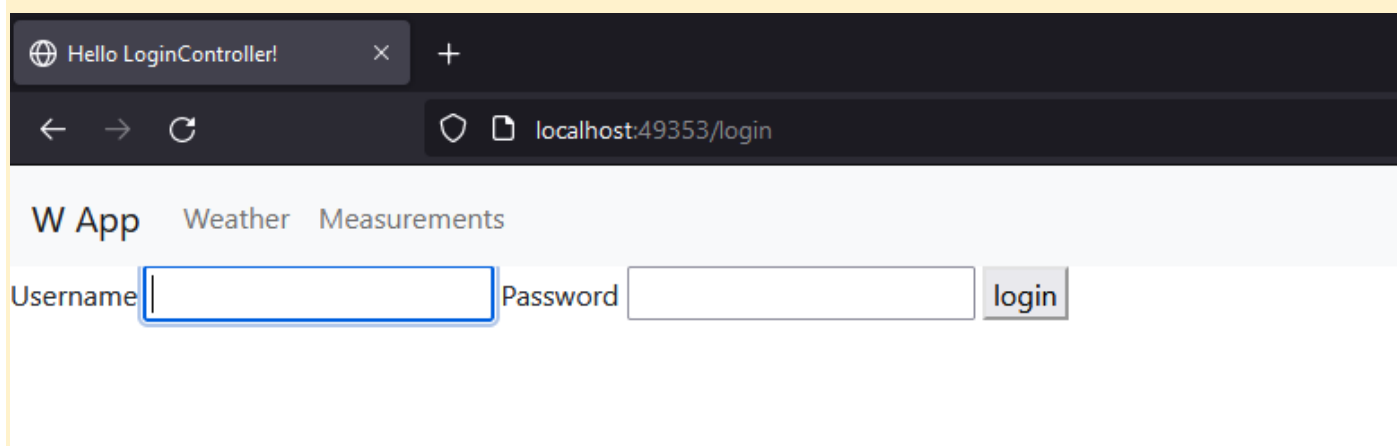
1  {% extends 'base.html.twig' %}
2
3  {% block title %}Login{% endblock %}
4
5  {% block body %}
6      {% if error %}
7          <div>{{ error.messageKey|trans(error.messageData, 'security') }}</div>
8      {% endif %}
9
10     <form action="{{ path('app_login') }}" method="post">
11         <label for="username">Username:</label>
12         <input type="text" id="username" name="_username" value="{{ last_username }}" />
13
14         <label for="password">Password:</label>
15         <input type="password" id="password" name="_password" />
16
17         {# If you want to control the URL the user is redirected to on success
18         <input type="hidden" name="_target_path" value="/admin" /> #}
19
20         <button type="submit">login</button>
21     </form>
22 {% endblock %}

```

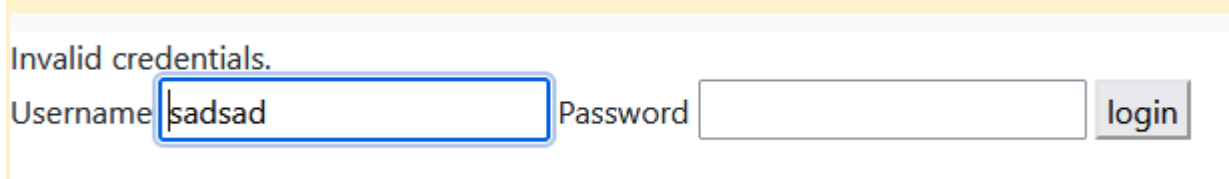
Po wejściu na stronę z ograniczeniem dostępu wyświetli się teraz formularz logowania:



Wstaw zrzut ekranu strony z formularzem logowania:



Podaj błędne dane logowania. Wstaw zrzut ekranu strony logowania z wyświetlonymi informacjami o błędach:



Zaloguj się na użytkownika, którego dane zapisane są w bazie danych. W pasku profilera kliknij na nazwę użytkownika. Otworzy się panel Security.

Wstaw zrzut ekranu panelu Security z danymi zalogowanego użytkownika:

Symfony Profiler

302 redirect from (POST) @app_login (50e3c2)

ERROR 403 Forbidden

GET http://localhost:49353/location/

← Browse referrer URL IP: ::1 Profiled on: November 27, 2023 at 6:38:00 PM Token: 78692b

→ Forwarded to ErrorController (bab3a1)

Security

Token Firewall Listeners Authenticators Access Decision

Username admin Authenticated ✓

Property	Value
Roles	["ROLE_ADMIN" "ROLE_USER"]
Inherited Roles	none
Token	Symfony\Component\Security\Core\Authentication\Token\UsernamePasswordToken {#633 ▾ -user: App\Entity\User {#492 ...} -roleNames: [] -attributes: [] -firewallName: "main" }

Punkty:

0

1

WYLOGOWYWANIE

Obecnie po zalogowaniu jedyną opcją wylogowania użytkownika jest skasowanie ciasteczka PHPSESSID. Dodanie logowania w Symfony jest proste. Wystarczy dodać nową ścieżkę do routes.yaml:

```
app_logout:
    path: /logout
    methods: GET
```

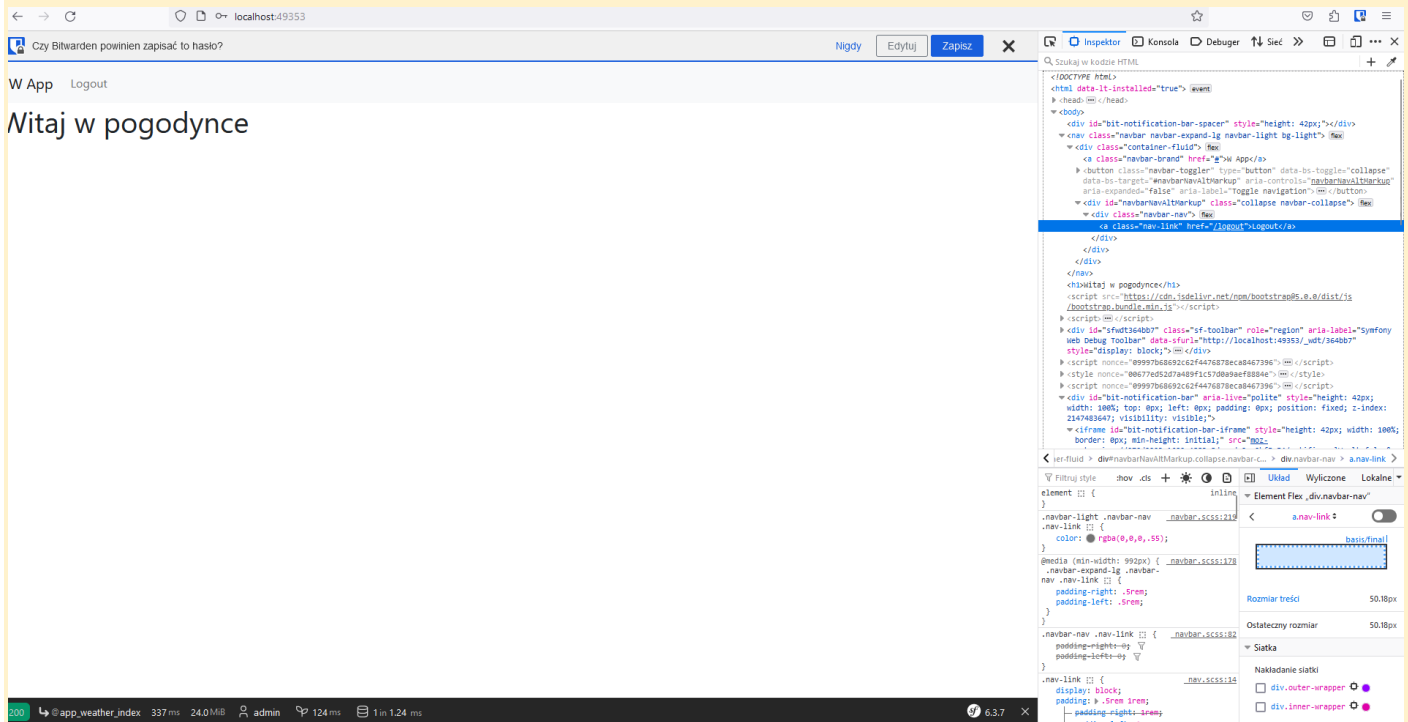
oraz zmodyfikować security.yaml:

main:	16	16	main:
lazy: true	17	17	lazy: true
provider: app_user_provider	18	18	provider: app_user_provider
http_basic:	>> 19	19	form_login:
realm: 'My Secured Area'	20	20	login_path: app_login
	21	21	check_path: app_login
# activate different ways to authenticate	22	22	logout:
# https://symfony.com/doc/current/security.html	23	23	path: app_logout

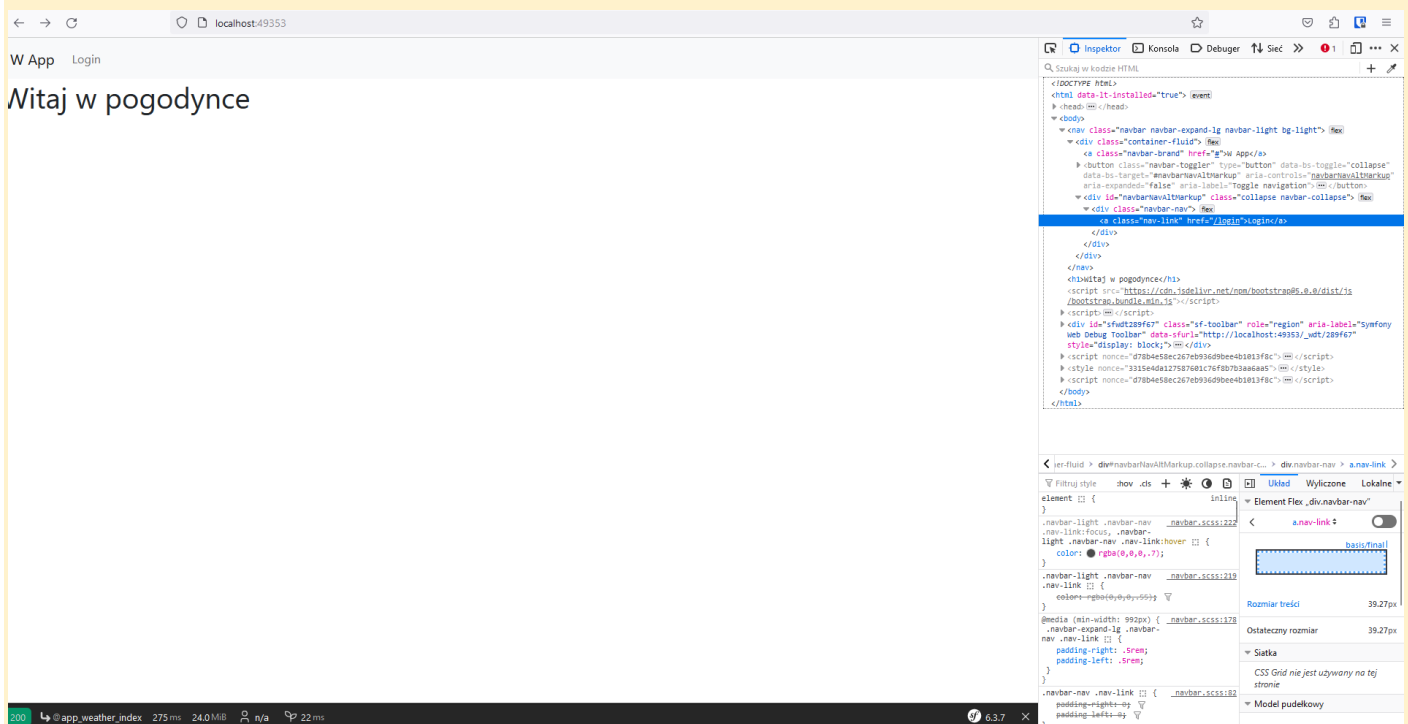
Na koniec należy dodać link do logowania / wylogowywania do pliku szablonu base.html.twig:

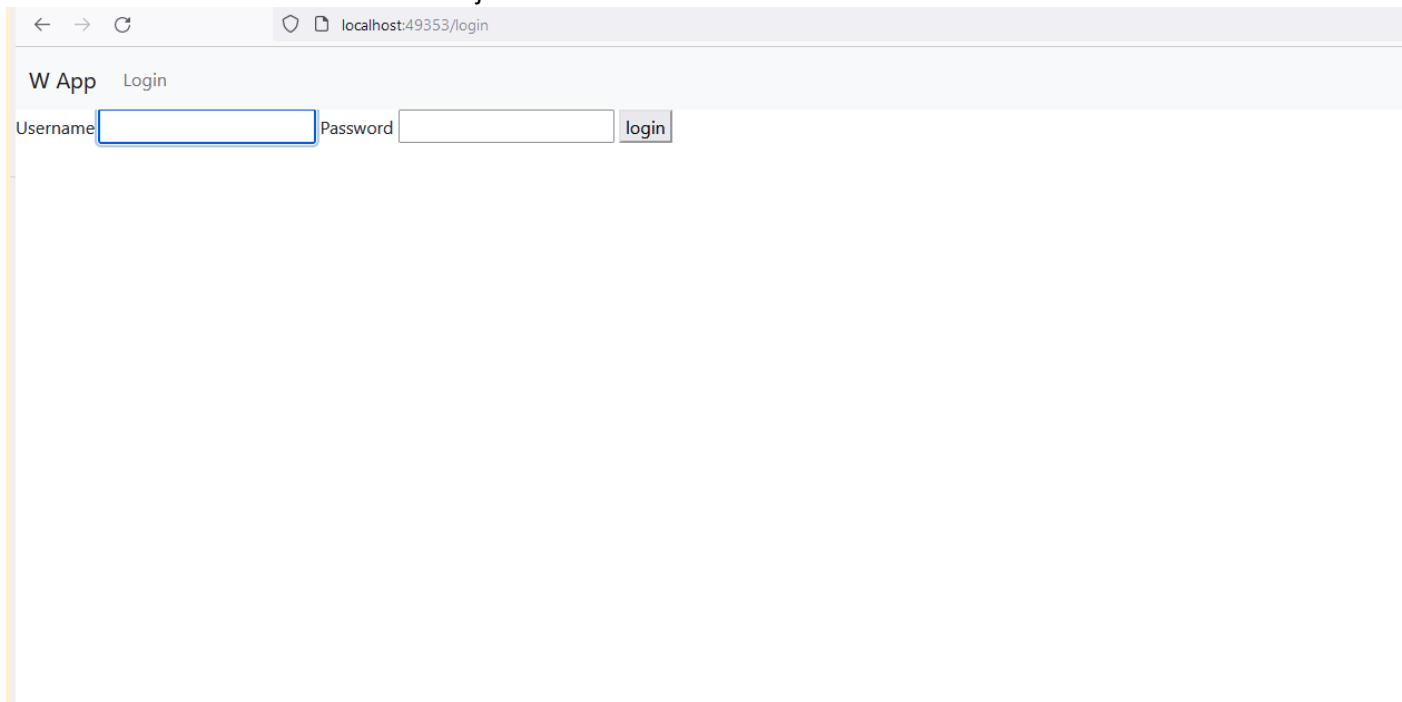
```
{% if is_granted('ROLE_USER') %}
    <a href="{% path('app_logout') %}">Logout</a>
{% else %}
    <a href="{% path('app_login') %}">Login</a>
{% endif %}
```

Wstaw zrzut ekranu strony z widocznym linkiem wylogowywania:



Wstaw zrzut ekranu strony z widocznym linkiem logowania:





Wstaw zrzut fragmentu kodu TWIG odpowiedzialnego za wyświetlanie linku logowania / wylogowywania:

```

... security.yaml M routes.yaml M WeatherController.php M base.html.twig M MeasurementController.php M welcome.html.twig U LoginContr
pogodynka > templates > base.html.twig
11 {% block stylesheets %}
12 {% endblock %}
13 </head>
14
15 <body>
16 <nav class="navbar navbar-expand-lg navbar-light bg-light">
17 <div class="container-fluid">
18 <a class="navbar-brand" href="#">W App</a>
19 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavAltMarkup" aria-controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle navigation">
20 <span class="navbar-toggler-icon"></span>
21 </button>
22 <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
23 <div class="navbar-nav">
24     {% if is_granted('ROLE_MEASUREMENT_INDEX') and is_granted('ROLE_LOCATION_INDEX') %}
25     <a class="nav-link" aria-current="page" href="{{ path('app_location_index') }}">Weather</a>
26     {% endif %}
27     {% if is_granted('ROLE_MEASUREMENT_INDEX') %}
28     <a class="nav-link" href="{{ path('app_measurement_index') }}">Measurements</a>
29     {% endif %}
30     {% if is_granted('ROLE_LOCATION_INDEX') %}
31     <a class="nav-link" href="{{ path('app_location_index') }}">Locations</a>
32     {% endif %}
33     ..... {% if is_granted('ROLE_USER') %} You, teraz * Uncommitted changes
34     ..... <a class="nav-link" href="{{ path('app_logout') }}">Logout</a>
35     ..... {% else %}
36     ..... <a class="nav-link" href="{{ path('app_login') }}">Login</a>
37     ..... {% endif %}
38 </div>
39 </div>
40 </div>
41 </nav>
42 {% block body %}
43 {% endblock %}
44 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0/dist/js/bootstrap.bundle.min.js"></script>
45 </script>

```

Wstaw zrzut ekranu konfiguracji security.yaml – fragment dotyczący wylogowywania:

HIERARCHIA RÓL

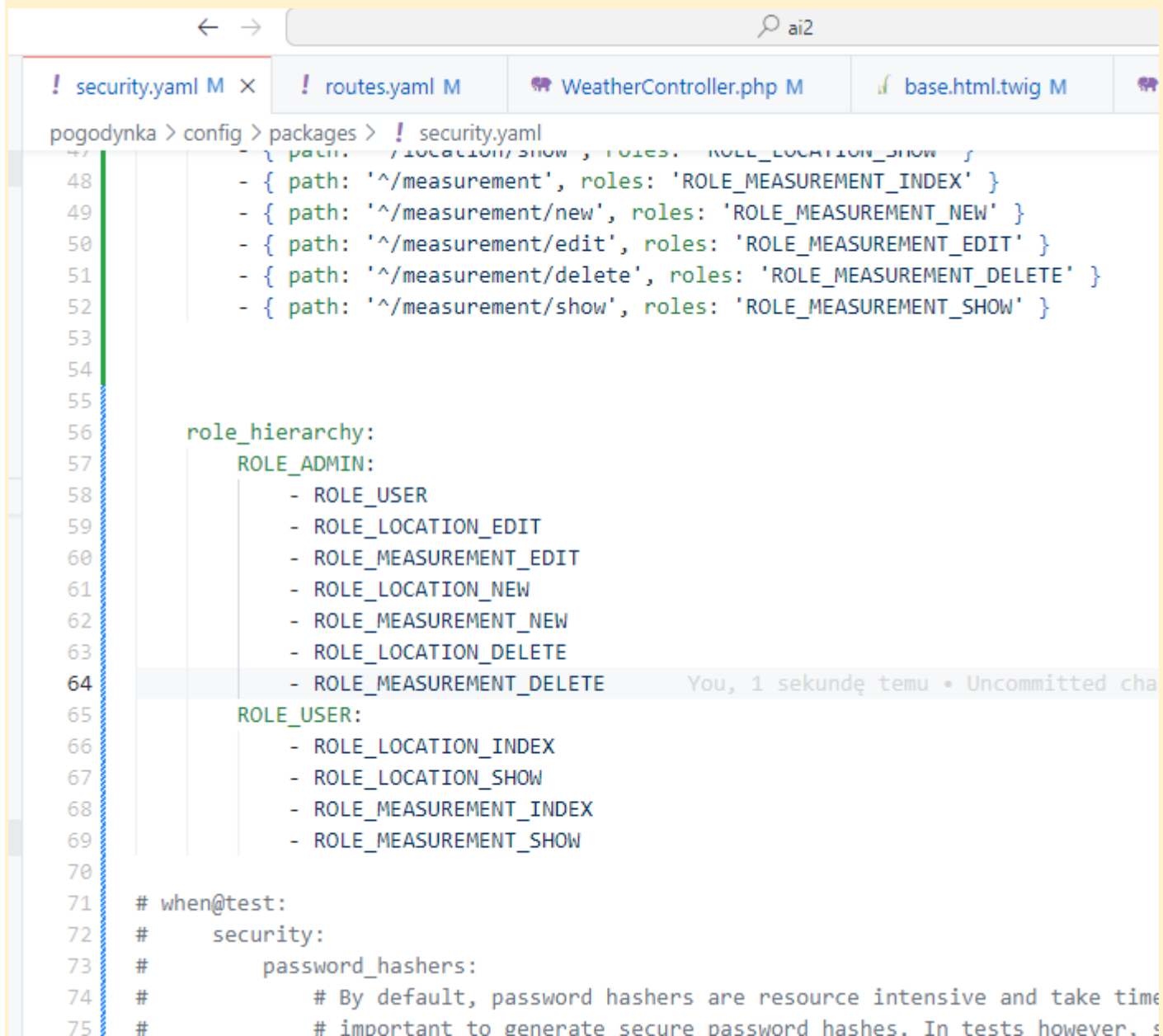
access_control:	30	32	access_control:
- { path: ^/, roles: PUBLIC_ACCESS }	31	33	- { path: ^/, roles: PUBLIC_ACCESS }
	>> 32	34	
!@test:	33	35	role_hierarchy:
security:	34	36	ROLE_ADMIN:
password_hashers:	35	37	- ROLE_USER
# By default, password hashers are resour	36	38	- ROLE_LOCATION_EDIT
# important to generate secure password I	37	39	ROLE_USER:
# are not important, waste resources and	38	40	- ROLE_LOCATION_INDEX

Wprowadź zmiany i wejdź na stronę /location:

Strona 17 z 21

Widoczna jest lista lokalizacji, a przy każdej z nich link do edycji. Nie ma natomiast linku do tworzenia nowej lokalizacji. Wejście na strony pomiarów zakończy się wyświetleniem błędu braku dostępu.

Uzupełnij hierarchię ról dla roli ROLE_ADMIN i dla ROLE_USER, tak aby obsługiwać wszystkie role obu kontrolerów Location i Measurement. Wstaw zrzut ekranu odpowiedniego fragmentu security.yaml:



```

48 - { path: '^/measurement', roles: 'ROLE_MEASUREMENT_INDEX' }
49 - { path: '^/measurement/new', roles: 'ROLE_MEASUREMENT_NEW' }
50 - { path: '^/measurement/edit', roles: 'ROLE_MEASUREMENT_EDIT' }
51 - { path: '^/measurement/delete', roles: 'ROLE_MEASUREMENT_DELETE' }
52 - { path: '^/measurement/show', roles: 'ROLE_MEASUREMENT_SHOW' }
53
54
55
56 role_hierarchy:
57   ROLE_ADMIN:
58     - ROLE_USER
59     - ROLE_LOCATION_EDIT
60     - ROLE_MEASUREMENT_EDIT
61     - ROLE_LOCATION_NEW
62     - ROLE_MEASUREMENT_NEW
63     - ROLE_LOCATION_DELETE
64     - ROLE_MEASUREMENT_DELETE
65   ROLE_USER:
66     - ROLE_LOCATION_INDEX
67     - ROLE_LOCATION_SHOW
68     - ROLE_MEASUREMENT_INDEX
69     - ROLE_MEASUREMENT_SHOW
70
71 # when@test:
72 #   security:
73 #     password_hashers:
74 #       # By default, password hashers are resource intensive and take time
75 #       # important to generate secure password hashes. In tests however, s

```

Utwórz nowego użytkownika, z innym zestawem uprawnień (ROLE_USER? Inna, nowa rola użytkownika?). Wstaw zrzut ekranu listy lokalizacji dla pierwszego użytkownika i dla drugiego użytkownika. Upewnij się, że poziom uprawnień jest różny, przez co różnią się dostępne na stronie akcje:

SQLite



SQL ▾

< 1 / 1 > 1 - 2 of 2

id	username	roles	password
9	admin	["ROLE_ADMIN"]	\$2y\$13\$uQ5fiPBbsbhIdaKdqsZkz.E.YeFWZK8Wyx5kyjPvxjZAXbEQcLL2i
10	user	["ROLE_USER"]	\$2y\$13\$uQ5fiPBbsbhIdaKdqsZkz.E.YeFWZK8Wyx5kyjPvxjZAXbEQcLL2i

[←](#) [→](#) [↻](#)[🔍](#) localhost:49353/location/W App [Weather](#) [Measurements](#) [Locations](#) [Logout](#)

Location Index

Id	City	Country	Latitude	Logitude	actions
1	Szczecin	PL	11	14.553	show edit
2	Police	PL	53.5521	14.5718	show edit
5	PoznanDDD	PL	11	10	show edit
6	d	PL	212	211	show edit

200 @app_location_index 509 ms 26.0 MiB [👤](#) admin [🔧](#) 17 ms [📄](#) 2 in 1.57 ms[🔍](#) 6.3.7

Measurement index

Id	Date	Celsius	actions
2	2012-01-01	23	<button>show</button>
3	2023-11-02	23.4	<button>show</button>
4	2023-11-03	24	<button>show</button>
5	2023-11-02	25	<button>show</button>

200 @app_measurement_index 648 ms 26.0 MiB user 18 ms 2 in 1.70 ms 6.3.7 X

Punkty:

0

1

COMMIT PROJEKTU DO GIT

Zacommittuj zmiany. Wyślij zmiany do repozytorium (push). Upewnij się, czy wszystko dobrze się wysłało. Jeśli tak, to z poziomu przeglądarki utwórz branch o nazwie `lab-e` na podstawie głównej gałęzi kodu.

Podaj link do brancha `lab-e` w swoim repozytorium:

...link, np. <https://github.com/ideaspot-pl/ai2-pogodynka-202310/tree/lab-e...>

<https://github.com/Michaldariusznowakowski/ai2/tree/lab-e>

PODSUMOWANIE

W kilku zdaniach podsumuj zdobyte podczas tego laboratorium umiejętności.

Podczas laboratorium, nauczyłem się tworzenia użytkowników i ról, implementacji mechanizmu uwierzytelniania, blokowania dostępu do niektórych widoków na podstawie ról, a także utworzenia panelu logowania i opcji wylogowania.

Zweryfikuj kompletność sprawozdania. Utwórz PDF i wyślij w terminie.