

Politechnika Białostocka Bezpieczeństwo Aplikacji Internetowych	Prowadzący: dr inż. Maciej Brzozowski
Temat: Generowanie haseł. 1. Dominik Adrian Kruk 2. Mateusz Matocha	Ocena:

1. Specyfikacja środowiska i technologii:

Język programowania: Java EE version 1.8.171

Baza danych: Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit
Production

Technologie: Spring 4.0; Thymeleaf 3.0.9; Hibernate 5.2; HTML 5;

System Operacyjny: Windows 10 Home Edition Wersja 10.0.17134 Kompilacja 17134

Program do zarządzania bazą danych: Oracle SQL Developer Version 17.3.0.271

Środowisko programistyczne(IDE): IntelliJ IDEA 2017.3.2 (Ultimate Edition)

Build #IU-173.4127.27, built on December 25, 2017


2. Wykonanie zadań:

- 1) Logowanie przy użyciu losowych wybranych fragmentów haseł przechowywanych po stronie serwera z wykorzystaniem funkcji hashujących i soli z następującymi kryteriami:

- Długość od 8 do 16 znaków;
- Maksymalnie zapytanie o fragment składający się z połowy hasła ale nie mniej niż pięć znaków;
- Walidacja miejsca znaków;
- Zmiana obecnego hasła po zalogowaniu z dodatkową weryfikacją;
- Minimum 10 haseł cząstkowych;

4) Zmiana hasła:

Password

	1		3	4		6		8								
---	---	--	---	---	--	---	--	---	--	--	--	--	--	--	--	--

Hasło

Powtórz hasło

Zmień hasło

Testy:

1) Czy dla użytkownika zmienia się wymagane hasło cząstkowe po wykonaniu zapytania:

Login form

Login form

Login

user2

Dalej

Login

Password

2

4

5


8

9

Login

Login form

Login

 user2

Dalej

Login

[illegible]

3. Czy użytkownikowi zmienia się wymagane hasło po nieprawidłowym logowaniu:

pobranie maski:

Request:

first

Examples (0)

POST

http://localhost:8082/login

Params

Send

Save

Authorization

Headers

Body

Pre-request Script

Tests

form-data

x-www-form-urlencoded

raw

binary

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	username	user1			
	New key	Value	Description		

Response:

[Napisz](#)

[-post](#)

Login

Password

1			4	
6		8	9	

Login

Przesłanie nieprawidłowego hasła(prawidłowe to password1):

Secondlogin

Examples (0)

POST

localhost:8082/secondLoginStep

Params

Send

Save

Authorization

Headers

Body

Pre-request Script

Tests

form-data


x-www-form-urlencoded

raw

binary

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	p0	p			
<input type="checkbox"/>	p1	a			
<input type="checkbox"/>	p2	s			
<input checked="" type="checkbox"/>	p3	s			
<input type="checkbox"/>	p4	w			
<input checked="" type="checkbox"/>	p5	o			
<input type="checkbox"/>	p6	r			
<input checked="" type="checkbox"/>	p7	3			
<input checked="" type="checkbox"/>	p8	1			
<input type="checkbox"/>	p9				

Response:

[Napisz](#) 

[-post](#)


Login form

Nieprawidłowa nazwa użytkownika lub hasło

Login

Dalej

Ponowny Request po maskę:

 Pretty Raw Preview

Save Resp

[Napisz](#)

[-post](#)

Login

Password

1			4	
6		8	9	

czy użytkownikowi zmienia się maska po prawidłowym zalogowaniu:

Dla maski jak wyżej przestanie hasła:

POST localhost:8082/secondLoginStep Params Send Save

Authorization Headers Body Pre-request Script Tests Cookies Code

☒ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary

	Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/>	p0	p			
<input type="checkbox"/>	p1	a			
<input type="checkbox"/>	p2	s			
<input checked="" type="checkbox"/>	p3	s			
<input type="checkbox"/>	p4	w			
<input checked="" type="checkbox"/>	p5	o			
<input type="checkbox"/>	p6	r			
<input checked="" type="checkbox"/>	p7	d			
<input checked="" type="checkbox"/>	p8	1			

