# CYBERWEJ III FAZA

PROJEKT W RAMACH PRZEDMIOTÓW IOSR I SIUS

*Krzysztof Wilaszek*
*Tomasz Wójcik*
*Piotr Leśniak*

## SPIS TREŚCI

# SCENARIUSZE TESTOWE

## WSTĘP

Scenariusze testowe zostały stworzone na podstawie wymagań funkcjonalnych aplikacji dostępnych w dokumencie z II fazy systemu. Pokrywają one 100% wymagań funkcjonalnych, realizując każde co najmniej raz.

Na końcu tego rozdziału znajduje się tabelka z obecnym statusem zaliczonych testów funkcjonalnych. Obrazuje ona ilość zrealizowanych funkcjonalności w systemie.

## 1. ZARZĄDZANIE KONTEM UŻYTKOWNIKA

### 1.1 TWORZENIE KONTA

#### SCENARIUSZ TESTOWY

Aktor: Niezalogowany użytkownik

1. Przejdź na stronę logowania.
2. Wybierz opcję tworzenia nowego konta.
3. Wprowadź wartości dla wszystkich wymaganych pól(identyfikator, imię, nazwisko) i opcjonalnie wartości dla pozostałych pól.
4. Wprowadź hasło dostępu do konta.
5. Zatwierdź wprowadzone dane.
6. Zaloguj się do system używając wcześniej wprowadzonego loginu i hasła.

Oczekiwany rezultat: Użytkownik zalogował się do systemu. Wyświetlona została główna strona aplikacji. Po wejściu na stronę danych o zalogowanym użytkowniku pokazują się poprawne dane wcześniej wprowadzone na formularzu.

### 1.2 LOGOWANIE NA ISTNIEJĄCEGO UŻYTKOWNIKA

#### SCENARIUSZ TESTOWY

Aktor: Niezalogowany użytkownik

1. Przejdź na stronę logowania.
2. Wprowadź identyfikator istniejącego użytkownika i poprawne hasło.
3. Naciśnij przycisk logowania.

Oczekiwany rezultat: Użytkownik zalogował się do systemu. Wyświetlona została główna strona aplikacji.

## 1.3 LOGOWANIE NA NIEISTNIEJĄCEGO UŻYTKOWNIKA

### SCENARIUSZ TESTOWY

Aktor: Niezalogowany użytkownik

1. Przejdź na stronę logowania.
2. Wprowadź identyfikator nieistniejącego użytkownika.
3. Naciśnij przycisk logowania.

Oczekiwany rezultat: Użytkownik nie zalogował się do systemu. Wyświetlony został komunikat o nieistniejącym użytkowniku.

## 1.4 LOGOWANIE NA ISTNIEJĄCEGO UŻYTKOWNIKA Z NIEPOPRAWNYM HASŁEM

### SCENARIUSZ TESTOWY

Aktor: Niezalogowany użytkownik

1. Przejdź na stronę logowania.
2. Wprowadź identyfikator istniejącego użytkownika i błędne hasło
3. Naciśnij przycisk logowania

Oczekiwany rezultat: Użytkownik nie zalogował się do systemu. Wyświetlony został komunikat o niepoprawnym haśle.

## 1.5 WYLOGOWANIE SIĘ Z SYSTEMU

### SCENARIUSZ TESTOWY

Aktor: Zalogowany użytkownik

1. Naciśnij przycisk wylogowania się z aplikacji.

Oczekiwany rezultat: System wylogowuje użytkownika, i przenosi go na główną stronę logowania.

## 1.6 WYŚWIETLENIE GRUP UŻYTKOWNIKA

### SCENARIUSZ TESTOWY

Aktor: Zalogowany użytkownik należący do co najmniej jednej grupy.

1. Przejdź do widoku głównego aplikacji.

Oczekiwany rezultat: System wyświetla wszystkie grupy użytkownika – ich nazwy wraz z bilansem wpłat użytkownika względem grup.

# 2. ZARZĄDZANIE GRUPAMI

## 2.1 STWORZENIE GRUPY I ZAPROSZENIE UŻYTKOWNIKÓW

### SCENARIUSZ TESTOWY

Aktor: Zalogowany użytkownik.

1. Wybierz z menu opcję tworzenia nowej grupy.
2. Wprowadź nazwę grupy.
3. Wybierz użytkowników, których chcesz dodać do grupy.
4. Zaakceptuj utworzenie nowej grupy.

Oczekiwany rezultat: System utworzył nową grupę. Użytkownik, który dodawał grupę został automatycznie do niej dodany. Wejdź na główną stronę aplikacji i sprawdź czy na liście Twoich grup pojawiła się nowa grupa.

Zaloguj się na konta użytkowników których dodałeś do grupy i sprawdź czy każdy otrzymał notyfikację z zaproszeniem do grupy.

## 2.2 AKCEPTACJI ZAPROSZENIA DO GRUPY

### SCENARIUSZ TESTOWY

Aktor: Zalogowany użytkownik, który został zaproszony do grupy

1. Zatwierdź zaproszenie do grupy przez kliknięcie na nie w tabeli notyfikacje.

Oczekiwany rezultat: System dodał użytkownika do danej grupy. Przejdź na główną stronę aplikacji i zobacz czy grupa pojawiła się w Twoich grupach. Wejdź do widoku tej grupy i zobacz czy ten użytkownik pojawił się w grupie.

## 2.3 DODANIE CZŁONKA DO ISTNIEJĄCEJ GRUPY

### SCENARIUSZ TESTOWY

Aktor: Członek grupy

1. Przejdź do widoku grupy do której chcesz dodać użytkownika.
2. Wybierz opcję dodaj użytkownika.
3. Wyszukaj użytkownika, którego chcesz dodać.
4. Zatwierdź zaproszenie użytkownika do grupy.

Oczekiwany rezultat: System dodał użytkownika do istniejącej grupy. Zaloguj się jako zaproszony użytkownik i sprawdź czy dostał on notyfikację z zaproszeniem do grupy.

## 2.4 OPUSZCZENIE GRUPY

### SCENARIUSZ TESTOWY

Aktor: Członek grupy, który nie jest nic winien innym członkom grupy.

1. Przejdź do widoku grupy, którą chcesz opuścić.
2. Wybierz opcję wystąpienia z grupy.
3. Potwierdź swój wybór.

Oczekiwany rezultat: System usunął użytkownika z  grupy. Przejdź na główną stronę aplikacji i zobacz czy grupa zniknęła z widoku Twoich grup.

## 2.5 WYŚWIETLENIE PODSTAWOWYCH INFORMACJI O GRUPIE

### SCENARIUSZ TESTOWY

Aktor: Członek grupy wcześniej utworzonej grupy, z dodanymi użytkownikami i płatnościami.

1. Na głównej stronie aplikacji naciśnij na wybraną grupę.

Oczekiwany rezultat: System wyświetla informację o wybranej grupie. Sprawdź czy poprawna jest nazwa grupy, data jej utworzenia, liczba użytkowników oraz liczba płatności grupy.

## 2.6 WYŚWIETLENIE INFORMACJI O CZŁONKACH GRUPY

### SCENARIUSZ TESTOWY

Aktor: Członek grupy wcześniej utworzonej grupy, z dodanymi użytkownikami i płatnościami.

1. Na głównej stronie aplikacji naciśnij na wybraną grupę.

Oczekiwany rezultat: System wyświetla informację o wybranej grupie. Sprawdź czy wyświetleni zostali wszyscy członkowie grupy, informacje o ich aktualnym statusie względem grupy oraz data dołączenia do grupy.

### 2.7 WYŚWIETLENIE INFORMACJI OZAREJESTROWANYCH PŁATNOŚCIACH W GRUPIE

#### SCENARIUSZ TESTOWY

Aktor: Członek grupy wcześniej utworzonej grupy, z dodanymi użytkownikami i płatnościami.

1. Na głównej stronie aplikacji naciśnij na wybraną grupę.

Oczekiwany rezultat: System wyświetla informację o wybranej grupie. Sprawdź czy wyświetlone wszystkie płatności zarejestrowanych w grupie, informacje o dacie ich utworzenia, ilości wydanych pieniędzy oraz użytkownikach biorących udział w płatności.

## 3. RACHUNKI I OPŁATY

### 3.1 DODANIE NOWEJ OPŁATY

#### SCENARIUSZ TESTOWY

Aktor: Członek grupy

1. Wybierz opcję rejestrowania nowej opłaty.
2. Wprowadź nazwę opłaty i wybierz grupę w ramach której zostanie zarejestrowana opłata.
3. Dodaj pozycję do opłaty wprowadzając nazwę produktu, cenę i ilość oraz wybierając użytkowników którzy spożyli dany produkt. Przejdź do następnego ekranu.
4. Dodaj uczestników opłaty którzy zapłacili, wybierając użytkownika i wpisując kwotę którą zapłacił. Przejdź do następnego ekranu.
5. Zaakceptuj dodanie nowej opłaty.

Oczekiwany rezultat: System rejestruje wprowadzoną opłatę. Wejdź na główny ekran aplikacji i zobacz czy w tabeli ostatnich płatności została dodana płatność z poprawną ilością wydanych przez użytkownika pieniędzy. Wejdź w widok grupy w której zarejestrowałeś płatność i zobacz czy została dodana opłata z poprawnymi informacjami.

### 3.2 DODANIE NOWEJ POZYCJI DO OPŁATY

#### SCENARIUSZ TESTOWY

Aktor: Członek grupy z zarejestrowaną płatnością.

1. Wejdź na widok płatności do której chcesz dodać opłatę.
2. Wprowadź nazwę produktu, cenę i ilość oraz wybierz użytkowników którzy spożyli dany produkt.

3. Zatwierdź dodawanie pozycji.

Oczekiwany rezultat: System rejestruje wprowadzoną pozycję do opłaty. Wejdź na widok opłaty i zobacz czy została dodana nowa pozycja z dobrze wypełnionymi informacjami o cenie, ilości i konsumentach. Sprawdź czy status konsumentów pozycji został pomniejszony o odpowiednią wartość w opłacie.

## 3.3 WYŚWIETLENIE INFORMACJI O UCZESTNIKACH OPŁATY

### SCENARIUSZ TESTOWY

Aktor: Użytkownik zarejestrowany jako uczestnik opłaty

1. Przejdź do widoku zarejestrowanej opłaty.

Oczekiwany rezultat: Na widoku płatności prezentowane są informacje o uczestnikach opłaty. Sprawdź czy zgadza się liczba skonsumowanych produktów, zapłacona kwota oraz aktualny status w opłacie.

## 3.4 WYŚWIETLENIE INFORMACJI O POZYCJACH OPŁATY

### SCENARIUSZ TESTOWY

Aktor: Użytkownik zarejestrowany jako uczestnik opłaty

1. Przejdź do widoku zarejestrowanej opłaty.

Oczekiwany rezultat: Na widoku płatności prezentowane są informacje o pozycjach opłaty. Sprawdź czy zgadza się ich liczba a także informacje o cenie, ilości i konsumentach pozycji.

## 3.5 ZGŁOSZENIE SPŁATY DŁUGU

### SCENARIUSZ TESTOWY

Aktor: Członek grupy z niedopłatą

1. Przejdź do widoku rejestracji spłaty długu.
2. Wpisz kwotę spłaty.

3. Wybierz grupę.
4. Wybierz użytkownika dla którego zostanie zarejestrowana spłata.
5. Potwierdź rejestracji spłaty.

Oczekiwany rezultat: System wysłał do użytkownika informację o spłacie długu. Zaloguj się na konto użytkownika i zobacz czy w tabelce notyfikacji pojawiła się odpowiednia wartość.

## 3.6 POTWIERDZENIE SPŁATY DŁUGU

### SCENARIUSZ TESTOWY

Aktor: Członek grupy z nadpłatą, któremu został spłacony dług

1. Zaloguj się na użytkownika, któremu spłacony został dług.
2. Z tabeli z notyfikacjami wybierz tą, która opisuje spłatę długu i zatwierdź ją.

Oczekiwany rezultat: System zarejestrował spłatę długu. Zobacz na widoku grupy czy zmieniły się statusy obydwu użytkowników (spłacającego i spłacanego) względem grupy.

## STAN PRZEPROWADZONYCH TESTÓW W KOLEJNYCH WERSJACH

| Wersja aplikacji | 1.1 | 1.2 | 1.3 | 1.4 | 1.5 | 1.6 | 2.1 | 2.2 | 2.3 | 2.4 | 2.5 | 2.6 | 2.7 | 3.1 | 3.2 | 3.3 | 3.4 | 3.5 | 3.6 | Passed | Failed | Percent of realization |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.0.1 | | | | | | | | | | | | | | | | | | | | 9 | 10 | 47 % |
| 0.0.2 | | | | | | | | | | | | | | | | | | | | 17 | 2 | 89 % |

W wersji 0.0.2 oddawanej na koniec trzeciej fazy zabrakło funkcjonalności dodania członka do istniejącej grupy oraz opuszczenia grupy. Przeprowadzenie pozostałych scenariuszy testowych dało wynik pozytywny co oznacza, że większość wymagań funkcjonalnych została zrealizowana.

# BEZPIECZEŃSTWO APLIKACJI

## MECHANIZMY BEZPIECZEŃSTWA I AUTENTYKACJI WBUDOWANE W APLIKACJĘ

### ZREALIZOWANE

Aby dostać się do aplikacji należy zalogować się podając poprawny login i hasło istniejącego w bazie użytkownika na głównej stronie logowania do aplikacji. W przypadku próby logowania na nieistniejącego użytkownika lub podania błędnego hasła system wyświetli odpowiednią informację.

Hasła użytkowników przechowywane są w bazie danych w postaci zaszyfrowanej. Szyfrowanie odbywa się tuż po wprowadzenia hasła, dlatego nigdy nie jest ono w przesyłane w postaci plain textu.

Jeżeli użytkownik nie jest zalogowany do aplikacji, każda próba wejścia na inny ekran niż ekran logowania lub zakładania konta użytkownika spowoduje przekserowanie na stronę logowania.  Mechanizm ten został zrealizowany poprzez wpięcie się w odpowiedni etap renderowania strony przez JSF i sprawdzenie warunku zalogowania użytkownika.

### DO ZREALIZOWANIA

Kolejnym krokiem w zabezpieczeniu aplikacji, byłoby stworzenie ról dla użytkowników i ograniczenie im widoków zgodnie z tymi rolami. Wtedy, konkretne informacje o płatnościach, użytkownikach lub grupach widoczne by były tylko dla członków tej samej grupy. Mechanizm ten można zrealizować przy użyciu części Frameworku Spring – SpringSecurity lub podpinając zewnętrzny system taki jak np. LDAP.

## BEZPIECZEŃSTWO INSTALACJI KONTENERA SERWLETÓW TOMCAT

### ZABEZPIECZENIE INSTALACJI TOMCATA:

- Usunięcie domyślnie zainstalowanych aplikacji, tak aby nie udostępniać informacji o środowisku uruchamiania.
- Zmiana domyślnej komendy wyłączenia serwera, wpisując `<server port="8098" shutdown="goingdown">` w pliku conf/server.xml.
- Uruchamianie Tomcata z konta specjalnie przygotowanego użytkownika, aby szczegółowo wyspecyfikować uprawnienia serwera odnośnie zasobów systemu operacyjnego.
- Zabezpieczenie systemu plików.

- Zabezpieczenie JVM, pozwalające definiować uprawnienia aplikacji do narażonych na ataki zasobów.
- Uruchomienie Tomcata w trybie security możliwe jest poprzez dodanie opcji –security w argumentach startowych serwera. Mechanizm tez pozwala definiować uprawnienia aplikacji do określonych zasobów w plikach konfiguracyjnych. Uprawnienia określane są w pliku conf/catalina.policy.

### TECHNIKI ZABEZPIECZANIA APLIKACJI WEBOWYCH URUCHOMIONYCH NA TOMCACIE:

- Autentykacja – Tomcat wykorzystuje obszary bezpieczeństwa przechowujące dane autentykacji użytkowników aplikacji. Istnieją 4 mechanizmy autentykacji - BASE, DIGEST, FORM, HTTPS client certificate. Mechanizmy autentykacji konfigurowane są w pliku web.xml. Użytkownicy są przyporządkowani do określonych ról, którym nadawane są uprawnienia.
- Szyfrowanie danych. Tomcat ma ustawiony konektor https umożliwiający bezpieczne połączenia klietna z serwerem przy użyciu SSL.
- Wyłączenie DefaultServlet. Jest to servlet obsługujący wszystkie próby połączeń do nieistniejących zasobów. Umożliwienie jego działania stanowi punkt zagrożenia aplikacji, ponieważ pozwala na dostęp do informacji o wewnętrznych mechanizmach działania aplikacji.
- Filtrowanie lub odrzucanie zapytań od określonych klientów aplikacji.

## BEZPIECZEŃSTWO APLIKACJI WEB

### ZABEZPIECZENIA ŚRODOWISKA URUCHOMIENIOWEGO

W przypadku środowiska uruchomieniowego(system operacyjny, Java, serwer aplikacji) należy przede wszystkim zadbać o regularne aktualizowanie oprogramowania, ze szczególnym uwzględnieniem wszystkich krytycznych poprawek bezpieczeństwa. Można rozważyć zastosowanie odpowiednich modułów podnoszących ogólny poziom bezpieczeństwa w systemie, np. SELinux.

### ZABEZPIECZENIA BAZY

Dostęp do bazy danych MySQL jest zabezpieczony hasłem. Jednak przechowywanie hasła w pliku konfiguracyjnym wykorzystywanym do połączenia z bazą, powoduje, że każdy kto ma dostęp do kodu źródłowego aplikacji ma również dostęp do bazy. Sugerowanym rozwiązaniem jest zastosowanie zewnętrznego modułu kontroli dostępu, wykorzystującego rozwiązania PAM, LDAP lub Kerberos.

## ZABEZPIECZENIA APLIKACJI

Warstwą na której poziom bezpieczeństwa mamy największy wpływ jest tworzona przez nas aplikacja. Potencjalnym miejscem ataków typu Cross-site scripting (XSS), Cross-site request forgery(CSRF) i SQL injection są formularze i co za tym idzie komponenty przetwarzające wejście użytkownika przesłane w postaci zapytań protokołu HTTP.

- **Cross-site scripting** - polega na wprowadzeniu na stronę fragmentu kodu, który wyświetlony przez innych użytkowników może spowodować wykonać w ich imieniu niepożądane akcje. JSF posiada domyślnie włączone podstawowe zabezpieczenie przeciwko XSS, polegające na zamianie specjalnych znaków HTML na encje(np. znak < jest zamieniany na <. Artykuły na temat bezpieczeństwa aplikacji napisanych w JSF, przestrzegają przed wypisywaniem wprowadzonych przez użytkownika wartości wewnątrz niektórych tagów lub wewnątrz atrybutów HTML. Zastosowanie walidacji poszczególnych pól, np. akceptacja tylko liter oraz cyfr, wprowadzenie maksymalnej długości wejścia, dodatkowo zwiększa poziom bezpieczeństwa.
- **Cross-site request forgery** - metoda opiera się na przesyłaniu do web aplikacji spreparowanych żądań pochodzących od zalogowanych użytkowników, odwiedzających przygotowane do ataku strony internetowe. Przeciwdziałanie takim atakom może polegać np. na umieszczeniu na stronie z formularzem losowego tokenu, dołączaniu go do requestów klienta i ponownej weryfikacji po stronie serwera.
- **SQL injection** - to wstrzykiwanie poprzez formularz kodu SQL, który ma na celu np. wyciągnięcie danych do których użytkownik nie ma uprawnień, bądź usunięcie danych. W tworzonej przez nas aplikacji, do persystencji danych wykorzystany jest framework Hibernate, który praktycznie zabezpiecza przed atakami SQL injection w przypadku niekorzystania z natywnych zapytań SQL, ale jest narażony na wstrzyknięcie kodu w języku HQL.
- **Autoryzacja użytkowników** - w kolejnej fazie planujemy zintegrowanie logowania do aplikacji z systemem LDAP, tymczasowo autoryzacja użytkowników opiera się na weryfikacji obliczonego hashu wprowadzonego hasła z hashem przechowywanym w bazie danych

Obecnie nasza aplikacja nie oferuje możliwości przesyłania plików na serwer, jeżeli w przyszłości miałaby być wprowadzona np. funkcjonalność dodawania avatarów użytkowników, należałoby rozszerzyć rozważania związane z bezpieczeństwem również o ten aspekt.

## PROPOZYCJE DZIAŁAŃ MAJĄCYCH NA CELU ZWIĘKSZENIE POZIOMU BEZPIECZEŃSTWA

W środowisku produkcyjnym proponujemy zastosowanie następujących narzędzi w celu zwiększenia poziomu bezpieczeństwa:

- Użycie LDAP do autoryzacji użytkowników
- Użycie PAM, LDAP bądź Kerberosa do autoryzacji dostępu do bazy danych

- Wykorzystanie odpowiednio zabezpieczonego systemu operacyjnego i kontenera serwletów

W odniesieniu do implementacji aplikacji będzie konieczna:

- walidacja wszystkich pól wprowadzanych przez użytkownika
- użycie tokenu zabezpieczającego przed CSRF

Oprócz tego można rozważyć zastosowanie narzędzi testujących aplikację pod kątem podatności na ataki, np. skaner Nessus.

## MAKIETA

Przed implementacją widoków aplikacji została stworzona oraz zaprezentowana klientowi makieta, pokazująca wszystkie funkcjonalności udostępniane przez Cyberwej.

Makieta została stworzona w narzędziu Axure RP Pro 6.5.

Makieta znajduje się pod adresem: http://student.agh.edu.pl/~lpiotr/Makieta0.6/