

## Porównanie wybranych bibliotek do tworzenia graficznego interfejsu internetowych aplikacji JEE

Piotr Paweł Suwała\*, Beata Pańczyk

Politechnika Lubelska, Instytut Informatyki, Nadbystrzycka 36B, 20-618 Lublin, Polska

**Streszczenie.** W środowisku Java Enterprise Edition podstawową technologią tworzenia graficznego interfejsu użytkownika (GUI) aplikacji internetowych jest JavaServer Pages (JSP). Możliwości oferowane przez strony JSP nie są jednak wystarczające, co spowodowało powstanie wielu rozszerzeń wspomagających programistę w szybkim tworzeniu nowoczesnych GUI. W artykule przedstawiono wyniki porównania bibliotek: JavaServer Faces, Rich Faces i Prime Faces.

**Słowa kluczowe:** JavaServer Faces; PrimeFaces; RichFaces; GUI

\* Autor do korespondencji.

Adres e-mail: suwcio@gmail.com

## Comparison of selected graphical web interface libraries used in JEE applications

Piotr Paweł Suwała\*, Beata Pańczyk

Institute of Computer Science, Lublin University of Technology, Nadbystrzycka 36B, 20-618 Lublin, Poland

**Abstract.** The basic technology for creating a graphical user interface (GUI) for Java Enterprise Edition web applications is JavaServer Pages (JSP). However, the capabilities offered by JSP are not sufficient, which has resulted in many extensions that assist the programmer in quickly creating modern GUI. The paper presents the results of comparison JavaServer Faces, Rich Faces and Prime Faces libraries.

**Keywords:** JavaServer Faces; PrimeFaces; RichFaces; GUI

\*Corresponding author.

E-mail address: suwcio@gmail.com

### 1. Wstęp

Bardzo istotną rzeczą podczas produkcji oprogramowania jest zapewnienie atrakcyjnego graficznego interfejsu użytkownika (GUI). Z punktu widzenia klienta to nie tylko funkcjonalność, ale w dużym stopniu nowoczesny wygląd i intuicyjny interfejs wpływa na pozytywny odbiór aplikacji.

Java Enterprise Edition (JEE) [1, 2, 3] jest rozbudowaną platformą programowania, umożliwiającą, między innymi, tworzenie aplikacji internetowych. W jej skład wchodzi wiele technologii pozwalających na szybkie i zautomatyzowane tworzenie oprogramowania. Dodatkowe rozszerzenia są tworzone i udostępniane przez firmy zewnętrzne (np. cieszący się popularnością Spring czy trochę już przestarzały Struts).

W środowisku JEE powstało wiele rozwiązań wspomagających budowę interesujących GUI. Najbardziej znanym sposobem obrazowania danych, są strony JavaServer Pages (JSP) [4]. Składają się one ze zwykłych znaczników HTML oraz znaczników specyficznych dla JSP, które odpowiadają za dynamiczne generowanie treści wynikowego dokumentu HTML.

Strony JSP okazały się znakomitą podstawą dla innych technologii generowania widoku. Bardzo dobrym przykładem jest tu JavaServer Faces (JSF) [5, 6]. Potrzeby rynku wymusiły rozwój kolejnych rozszerzeń umożliwiających tworzenie bardziej atrakcyjnych i nowoczesnych interfejsów użytkownika. Między innymi są to PrimeFaces [7] oraz RichFaces [8].

### 2. Cel badań

Celem badań była analiza funkcjonalności oraz porównanie najbardziej popularnych bibliotek ułatwiających proces tworzenia graficznego interfejsu użytkownika internetowych aplikacji JEE.

Tezę badawczą sformułowano następująco: *Dla aplikacji internetowych tworzonych w środowisku JEE można budować atrakcyjne graficznie i nowoczesne interfejsy użytkownika.*

### 3. Aplikacje testowe

Do badań zostały utworzone trzy aplikacje testowe o identycznej funkcjonalności, różniące się jedynie technologią zastosowaną do budowy interfejsu graficznego. Są to odpowiednio JavaServer Faces (JSF), PrimeFaces

i RichFaces. Wybór tych bibliotek został podyktowany ich popularnością.

W aplikacjach, poza bibliotekami do tworzenia GUI, wykorzystano:

- Java 7,
- serwer bazy danych MySQL 5.6.11 [9],
- phpMyAdmin,
- technologię ORM – Hibernate 4.2.2.

Testy zostały przeprowadzone na komputerze z systemem operacyjnym Windows 8, Intel Core i5-3210M 2 x 2.50 GHz, 8 GB RAM.

Aplikacja jest formą bloga umożliwiającą zarządzanie postami oraz użytkownikami. Stronę główną aplikacji testowej zrealizowanej w oparciu o JSF przedstawiono na rysunku 1.



Rys. 1. Strona główna aplikacji testowej (JSF)

#### 4. Analiza porównawcza bibliotek

Do porównania trzech wybranych bibliotek analizowano 7 kryteriów:

- liczbę komponentów,
- łatwość implementacji,
- podstawowe możliwości,
- funkcje niewystępujące w innych bibliotekach,
- dostęp do dokumentacji, wsparcie,
- wydajność,
- popularność.

##### 4.1. Liczba komponentów

Liczba dostępnych kontrolki GUI została określona na podstawie witryny każdej biblioteki. Witryny te prezentują wygląd, implementację oraz sposób zastosowania każdej kontrolki (ang. showcase).

W Tabeli 1 znajduje się liczba kategorii oraz komponentów dla poszczególnych bibliotek.

Tabela 1. Liczba kategorii i komponentów w poszczególnych bibliotekach

	JSF	PrimeFaces	RichFaces
Liczba kategorii	13	15	12
Liczba komponentów	81	177	68

W przypadku liczby komponentów najlepiej wypadła biblioteka PrimeFaces. Posiada ich najwięcej co z założenia pozwala na budowanie bardziej różnorodnych oraz ciekawych interfejsów użytkownika.

##### 4.2. Łatwość implementacji

Każdą z bibliotek można zaimplementować w aplikacji na dwa sposoby. Pierwszym z nich jest pobranie biblioteki w tzw. pliku JAR i umieszczenie go w katalogu zawierającym biblioteki w aplikacji. Drugi sposób polega na automatycznym ściąganiu biblioteki bezpośrednio do projektu przy pomocy narzędzia Maven.

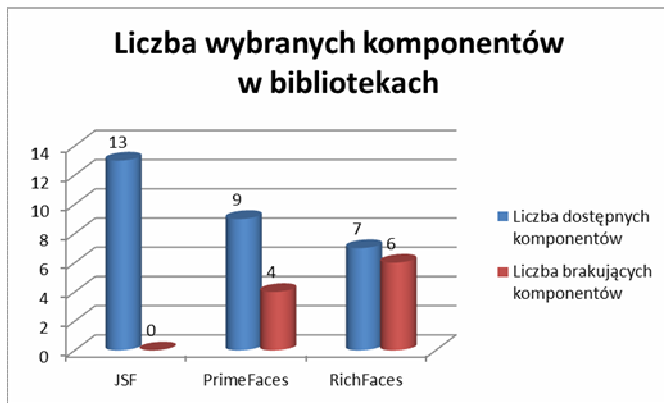
##### 4.3. Podstawowe możliwości

Każda z bibliotek posiada pewną liczbę komponentów, dzięki którym można stworzyć podstawowy interfejs użytkownika aplikacji webowej. Są to na przykład podstawowe kontrolki do wyświetlania i wprowadzania tekstu, lista rozwijana czy prezentacja tabelaryczna danych. W tabeli 2 przedstawiono wybrane znaczniki oraz wskazano ich dostępność w każdej z bibliotek.

Tabela 2. Dostępność podstawowych kontrolki w poszczególnych bibliotekach

Kontrolki	JSF	PrimeFaces	RichFaces
Przycisk polecenia	h:commandButton	p:commandButton	a4j:commandButton
Hiperłącze polecenia	h:commandLink	p:commandLink	a4j:commandLink
Tablica danych	h:dataTable	p:dataTable	rich:dataTable
Tekst wejściowy	h:inputText	p:inputText	brak
Plik wejściowy	h:inputFile	p:uploadFile	rich:fileUpload
Obraz graficzny	h:graphicImage	p:graphicImage	brak
Tekst wyjściowy	h:outputText	brak	brak
Panel siatka	h:panelGrid	p:panelGrid	brak
Wybór jednej wartości	h:selectOneMenu	brak	rich:select
Wybór pozycji	f:selectItem	brak	brak
	f:selectItems	brak	brak
Wiadomość oraz Wiadomości	h:message	p:message	rich:message
	h:messages	p:messages	rich:messages

Rysunek 2 obrazuje liczbę dostępnych oraz brakujących komponentów w poszczególnych bibliotekach.



Rys. 2. Liczba dostępnych oraz brakujących komponentów GUI (z grupy podstawowych kontrolki)

Biblioteka JSF zawiera większość podstawowych komponentów. Jednak warto zaznaczyć, że komponenty z biblioteki PrimeFaces wyglądają o wiele bardziej atrakcyjnie graficznie. Ich liczba jest nieco mniejsza, ale brakujące znaczniki można zastąpić tymi z JSF. Najslabiej prezentuje się biblioteka RichFaces, która posiada najmniejszą liczbę znaczników i przegrywa pod względem wyglądu z PrimeFaces.

#### 4.4. Funkcje niewystępujące w innych bibliotekach

##### JavaServer Faces

JSF udostępnia użytkownikom tzw. system szablonów. Pozwala on na stworzenie szkieletu strony WWW, a następnie tworzenia konkretnych stron na podstawie utworzonego szablonu.

Implementacja biblioteki JSF Mojarra [6] zawiera standardową bibliotekę znaczników Java. Funkcja ta jest jeszcze pozostałością po JSP i pozwala na obsługę podstawowych funkcji Java na stronach JSF.

##### PrimeFaces i RichFaces

Przewagą bibliotek PrimeFaces oraz RichFaces są większe możliwości tworzenia interfejsu użytkownika w oparciu o dodatkowe komponenty rozszerzające podstawowe funkcje (np. rozwijany panel, galeria, zegar, komponent pozwalający na wydruk zawartości innego komponentu, komponenty do tworzenia wykresów itp.). W tym wypadku dużo lepiej wypada PrimeFaces, która pozwala na osiągnięcie ciekawszych rezultatów poprzez ogromną różnorodność znaczników.

#### 4.5. Dokumentacja i wsparcie społecznościowe

Każda z prezentowanych bibliotek posiada dokumentację, wsparcie (np. w postaci forum lub FAQ) oraz społeczność zorganizowaną wokół danego frameworka. Tabela 3 prezentuje elementy dostępne w dokumentacji poszczególnych bibliotek.

Tabela 3. Elementy dostępne w dokumentacji w poszczególnych bibliotekach

	JSF	PrimeFaces	RichFaces
Instalacja	nie	tak	tak
Konfiguracja	nie	tak	tak
Opis kontrolki	tak	tak	tak
Przykłady	nie	tak	nie

Tabela 4 prezentuje rodzaje wsparcia dla poszczególnych bibliotek.

Tabela 4. Możliwości wsparcia w poszczególnych bibliotekach

	JSF	PrimeFaces	RichFaces
FAQ	tak	tak	tak
Forum	tak	tak	tak
Blog	nie	tak	tak
Get started	tak	tak	tak
Screencasts	nie	nie	tak

W tabeli 5 przedstawiono dostępność poszczególnych bibliotek na portalach społecznościowych.

Tabela 5. Dostępność na portalach społecznościowych

	JSF	PrimeFaces	RichFaces
Facebook	tak	tak	nie
Twitter	tak	tak	tak

Po przeanalizowaniu dostępnych materiałów na stronach poszczególnych bibliotek można stwierdzić, że najgorsze wsparcie posiada JSF. Najlepiej pod tym względem wypada biblioteka PrimeFaces, która oferuje szeroką dokumentację, przykłady zastosowań oraz wsparcie ze strony społeczności skupionej wokół technologii.

#### 4.6. Wydajność

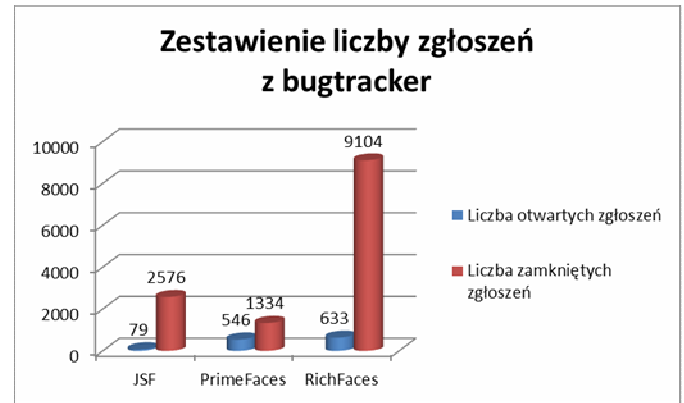
Mimo szybkich i wydajnych serwerów, to jak dany framework wpływa na czas generowania odpowiedzi na żądania klienta, jest jednym z najważniejszych kryteriów oceny. Wydajność bibliotek została zmierzona poprzez sprawdzenie całkowitego czasu ładowania aplikacji testowej przez przeglądarkę internetową. Posłużyła do tego aplikacja webowa PageScoring [10]. Jako serwer dla aplikacji testowych wykorzystano platformę OpenShift [11] w wersji darmowej. Udostępnia ona kontener dla aplikacji opartych o JEE.

Dla każdej biblioteki osobno, w aplikacji testowej wykonano 10 pomiarów prędkości ładowania strony głównej. Wyniki zostały przedstawione w tabeli 6.

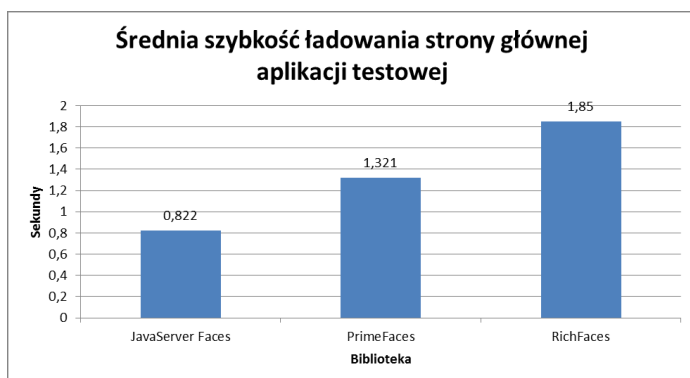
W celu lepszego zobrazowania wydajności aplikacji testowej na rysunku 3 przedstawiono wykres prezentujący średnie czasy pomiaru.

Tabela 6. Wyniki pomiarów szybkości ładowania strony głównej aplikacji

Numer pomiaru	JSF	PrimeFaces	RichFaces
1	0,75	1,35	2,05
2	0,67	1,33	1,53
3	0,63	1,57	3,81
4	0,69	1,5	1,79
5	0,63	1,31	1,87
6	1,44	1,19	1,4
7	0,85	1,27	1,39
8	0,69	1,27	1,39
9	1,04	1,22	1,58
10	0,83	1,2	1,69
<b>Średni czas [s]</b>	<b>0,822</b>	<b>1,321</b>	<b>1,85</b>



Rys. 4. Zestawienie liczby zgłoszeń z bugtracker



Rys. 3. Średnia szybkość ładowania strony głównej aplikacji testowej

Na wykresie wyraźnie widać, że aplikacja testowa z biblioteką JSF ładuje się najszybciej. Nieco wolniejszy czas ładowania został odnotowany przy testach z biblioteką PrimeFaces. Najwolniej strona ładowała się gdy została użyta biblioteka RichFaces.

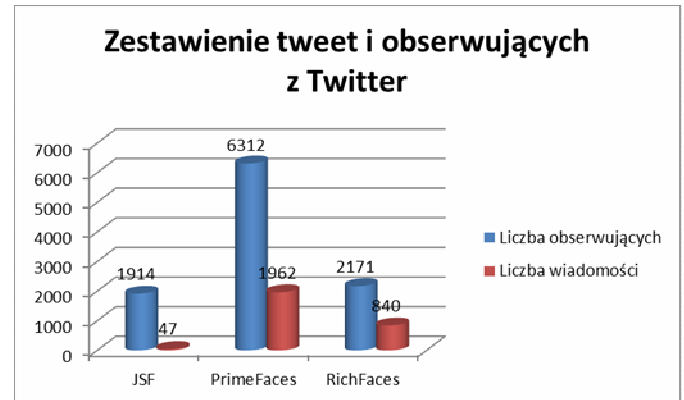
#### 4.7. Popularność

W celu zbadania popularności poszczególnych bibliotek, zastosowano dwa kryteria. Jest to:

- liczba aktualnie otwartych zadań na odpowiednich platformach typu 'bugtracker' (Rys. 4);
- liczba osób obserwujących stronę projektu oraz liczba wiadomości na portalu społecznościowym Twitter (Rys. 5).

Odnosnie pierwszego kryterium, tylko JSF i RichFaces używają tej samej platformy, ale służy ona jedynie do raportowania zgłoszeń o wykrytych problemach. Darmowa wersja PrimeFaces posługuje się platformą, która jest repozytorium, a dodatkowo pozwala na tworzenie zgłoszeń.

Z przedstawionych danych można wywnioskować, że największą popularnością cieszy się PrimeFaces. Świadczy o tym także duże zainteresowanie na portalu Twitter. Z liczby zadań zgłoszonych na portalach typu 'bugtracker' wynika, że intensywne prace trwają nad bibliotekami PrimeFaces oraz RichFaces. JSF nie jest już tak intensywnie rozwijany, stąd liczba zgłoszeń nie jest duża.



Rys. 5. Zestawienie z portalu Twitter

#### 4.8. Ocena bibliotek - podsumowanie

W tabeli 7 przedstawiono próbę oceny poszczególnych kryteriów w skali od 1 do 5. Nie stosowano wag dla poszczególnych kryteriów.

Tabela 7. Ocena bibliotek według wybranych kryteriów w skali 1 - 5

	JSF	PrimeFaces	RichFaces
Liczba komponentów	3	5	3
Łatwość implementacji	5	5	5
Podstawowe możliwości	5	4	3
Funkcje wyróżniające bibliotekę	3	5	3
Dokumentacja, wsparcie, społeczność	3	5	4
Wydajność	5	4	3
Popularność	3	5	4
<b>Średnia ocena końcowa</b>	<b>3,86</b>	<b>4,71</b>	<b>3,57</b>

Porównując wyniki z tabeli 7 widać, że najlepiej prezentuje się biblioteka PrimeFaces, następnie JavaServer Faces, na końcu zaś RichFaces.

## 5. Wnioski

Niewątpliwym atutem, który został zauważony podczas porównywania, jest to, że biblioteki nie wykluczają się nawzajem. W jednym projekcie istnieje możliwość używania komponentów z każdej analizowanej biblioteki. Pozwala to na ogromną elastyczność i dowolność stosowanych rozwiązań.

Z przeprowadzonej analizy wynika, że najlepszym wyborem do budowy w pełni funkcjonalnych aplikacji webowych z bogatym i przyjaznym interfejsem użytkownika jest połączenie możliwości JSF z PrimeFaces. JSF udostępnia podstawowe funkcje, niezbędne do utworzenia szkieletu strony internetowej, natomiast efektywniejszych wizualnie komponentów dostarcza PrimeFaces.

Jeśli programista musi się ograniczyć do jednego rozwiązania, to najlepszym wyborem jest JSF. Dwie pozostałe biblioteki nie oferują wszystkich podstawowych znaczników, bez których utworzenie kompletnej strony, nie jest możliwe.

Najmniej zachęcającą biblioteką okazała się RichFaces. O ile w pewnych aspektach (np. dokumentacja, wsparcie, funkcje wyróżniające) jest lepsza od JSF to jednak i tutaj PrimeFaces jest lepszym wyborem.

## Literatura

- [1] <http://www.oracle.com/technetwork/java/javaee/overview/index.html>, [10.04.2017].
- [2] E. Jendrock, R. Cervera – Navaroo, I. Evans, D. Gollapudi, K. Haase, W. Markito, C. Srivathsa, „Java EE 6 Zaawansowany przewodnik”, Wydanie IV, Helion, 2013.
- [3] K. Rychlicki – Kicior, „Java EE 6 Programowanie aplikacji WWW”, Wydanie II, Helion, 2015.
- [4] <http://www.oracle.com/technetwork/java/javaee/jsp/index.html>, [10.04.2017].
- [5] <http://www.oracle.com/technetwork/java/javaee/javaxserverfaces-139869.html>, [10.04.2017].
- [6] Mojarra JavaServer Faces – Oracle’s open source implementation of the JSF standard, <https://jaserverfaces.java.net/>, [20.02.2017].
- [7] PrimeFaces Ultimate UI Framework for Java EE, <http://www.primefaces.org/>, [20.02.2017].
- [8] RichFaces The next – generation JSF component framework by JBoss!, <http://richfaces.jboss.org/>, [20.02.2017].
- [9] K. Yank, „PHP i MySQL Witryna WWW oparta na bazie danych”, Wydanie III, Helion, 2005.
- [10] <http://www.pagescoring.com/website-speed-test/>, [20.02.2017].
- [11] <https://www.openshift.com/>, platforma testowa dla aplikacji JEE, [20.02.2017].