

Modelowanie BPMN

Laboratorium 5

Platforma Activiti

wersja 2.1

przygotował: dr inż. Radosław Adamus
Instytut Informatyki Stosowanej PŁ

Historia zmian

<i>Data</i>	<i>Wersja</i>	<i>Autor</i>	<i>Opis zmian</i>
24.03.2013	0.1	Radosław Adamus	Utworzenie dokumentu
05.04.2013	1.0	Radosław Adamus	Pierwsza oficjalna wersja dokumentu
08.04.2013	1.1	Radosław Adamus	Dodana informacja o edytorze (punkt 2.1) oraz rozwinięty punkt 1.7
11.04.2013	2.0	Radosław Adamus	Przeniesienie zadania 3 do laboratorium 6, modyfikacja celów laboratorium.
04.04.2014	2.1	Radosław Adamus	Poprawka związana z kompatybilnością instrukcji z wersją 5.15.1 Activiti.

Spis treści

Cel.....	3
1. Wprowadzenie.....	3
1.1 Activiti framework.....	3
1.2 Uruchomienie.....	3
Zadania.....	4
1. Analiza przykładowego procesu.....	4
2. Implementacja procesu	5
Użyteczne linki.....	6

Cel

1. Zapoznanie ze sposobem instalacji oraz podstawowymi funkcjonalnościami komponentów systemu Activiti.
2. Zapoznanie z możliwościami aplikacji Activiti Explorer w zakresie zarządzania wykonaniem instancji procesów oraz edytowania modeli wykonywalnych procesów.
3. Zrozumienie zależności pomiędzy definicją procesu w fazie modelowania, definicją wdrożoną oraz jej uruchomionymi instancjami.

1. Wprowadzenie

1.1 Activiti framework

Platforma Activiti to zestaw narzędzi umożliwiających rozwój aplikacji opartych na procesach biznesowych. Podstawowym komponentem Activiti jest silnik procesu (Activiti engine). Stanowi on implementację specyfikacji BPMN 2.0 a jego podstawowym zadaniem jest udostępnianie mechanizmów pozwalających na wykonywanie procesów. Oprócz silnika platforma udostępnia zestaw narzędzi dodatkowych do których należą:

Activiti Modeler – narzędzie do modelowania wykorzystujące notację BPMN w postaci aplikacji webowej.

Activiti Designer – wtyczka (plugin) do Eclipse IDE. Umożliwia projektowanie procesów zgodnych z BPMN 2.0 oraz wykorzystujących rozszerzenia Activiti (np. zadanie - usługa Java). Wtyczka wspomaga również proces testowania oraz budowy komponentów aplikacji, które można uruchamiać z wykorzystaniem silnika Activiti.

Activiti Explorer – Webowa aplikacja umożliwiająca zarządzanie silnikiem procesu. Do przykładowych funkcjonalności należą: uruchamianie procesu, odczytywanie listy zadań przypisanych zalogowanemu użytkownikowi, wdrażanie procesu, sprawdzanie statusu instancji procesu.

Activiti REST – Aplikacja webowa udostępniająca interfejs REST do silnika procesu (Activiti Engine). Stanowi ona domyślne API silnika.

1.2 Uruchomienie

Activiti dystrybuowany jest w postaci zestawu bibliotek oraz aplikacji webowych wymagających do swojego działania środowiska Java, serwera webowego (kontenera servletów JavaEE) oraz serwera baz danych. Domyślnie jako serwer webowy wykorzystywany jest Apache Tomcat a do składowania przykładowych procesów oraz ustawień wykorzystywana jest baza H2.

1. Dystrybucję można pobrać ze strony projektu (<http://www.activiti.org/download.html>) lub lokalnie <http://team.kis.p.lodz.pl:8080/svn/dydaktyka/bpmn/activiti/>.

2. Dla naszych potrzeb wystarczy z archiwum wydobyć plik `/wars/activiti-explorer.war`.

3. Plik `activiti-explorer.war` jest aplikacją webową, którą należy umieścić na serwerze. W tym celu należy zlokalizować na komputerze dystrybucję serwera Apache Tomcat (folder `c:/tools` lub c:/users/student/tools). Uruchomienie serwera odbywa się za pomocą pliku `bin/startup.bat`. Domyślnie serwer uruchomiony zostanie pod adresem <http://localhost:8080/>¹.

¹ W razie konfliktu należy dokonać edycji pliku `/conf/server.xml` i odnaleźć i zmodyfikować atrybut port w odpowiednim znaczniku `'connector'`

4. Umieszczenie aplikacji *activiti-explorer.war* na serwerze wymaga uruchomienia aplikacji Manager App (<http://localhost:8080/manager/html>). Aplikacja wymaga zalogowania a uprzednio odpowiednio skonfigurowanego serwera. Sprawdź w folderze dystrybucji Tomcat'a zawartość pliku */conf/tomcat-users.xml*. Plik powinien posiadać użytkownika z przypisaną rolą *manager-gui*. Przykładowa konfiguracja może wyglądać następująco:

```
<role rolename="manager-gui"/>
```

```
<user username="admin" password="admin" roles="manager-gui"/>
```

Po zalogowaniu do aplikacji menadżera należy znaleźć formularz umieszczania (ang. *deploy*) aplikacji na serwerze (**Deploy** → **WAR file to deploy**) i za jego pomocą umieścić *activiti-explorer.war*. Poprawne umieszczenie aplikacji *activiti-explorer* powinno udostępnić jej interfejs pod adresem <http://localhost:8080/activiti-explorer>. Do zalogowania należy podać nawet użytkownika oraz hasło: *'kermit'*.

5. Po poprawnym zalogowaniu zapoznaj się z panelami:

- a) Procesów (**Processes**). W panelu procesów dostępne są zakładki umożliwiające: przeglądanie instancji procesów (My Instances), zarządzanie listą możliwych do uruchomienia definicji procesów (Deployed process definitions) oraz dostęp do przestrzeni roboczej modeli (Model workspace) zawierającej definicje procesów w fazie modelowania. Z poziomu przestrzeni roboczej modeli możliwe jest tworzenie nowych definicji procesów, importowanie oraz edycja istniejących.
- b) Zarządzania (**Manage**) – dostęp wymaga uprawnień administracyjnych. Panel umożliwia m.in. zarządzanie użytkownikami, grupami, procesami.
- c) Zadań (**Tasks**). Panel pozwalający m.in. na wykonywanie zadań użytkownika, tworzenie nowych zadań oraz przyjmowanie istniejących do wykonania.

Zadania

1. Analiza przykładowego procesu

Znajdź w zakładce *Deployed process definition* (panel **Processes**) definicję procesu *Review sales lead*.

1.1 Uruchom proces. Sprawdź w zakładce *My instances* informacje o instancji procesu. Jaki jest stan procesu? Jakie są zmienne procesu?

1.2 Przejdź do panelu **Tasks** i ukończ zadanie procesu. Po zakończeniu przejdź do panelu procesów i sprawdź stan instancji procesu. Jaki jest stan procesu? Jakie są zmienne procesu?

1.3 Uruchom kolejną instancję procesu

1.4 Wróć do panelu zadań i sprawdź czy zalogowany użytkownik ma zadania do wykonania lub przyjęcia (zakładki *Inbox* oraz *Queued*).

1.5 Wywołaj ścieżkę obsługi błędu w pierwszym procesie.

1.6 Spróbuj zakończyć procesy.

1.7 Przekształć definicję procesu *Review sales lead* do wersji edytowalnej. Sprawdź:

- sposób przypisywania wykonawcy oraz kto jest przypisany jako wykonawca poszczególnych czynności.
- sposób definiowania formularzy czynności
- sposób definiowania warunków przepływu.

2. Implementacja procesu

- 2.1. Utwórz w edytorze *Yaoqiang BPMN* następujący (wstępny) model procesu:



Rysunek 1: Proces zamówienia książki

Nazwij proces *Zamówienie książki* i nadaj mu identyfikator *bookorder1*.

- 2.2. Zaimportuj definicję procesu do Activiti (panel **Processes** → zakładka *Model workspace*).

- 2.3. Dokonaj edycji procesu i zmodyfikuj go w następujący sposób:

2.3.1. Dodaj zadanie użytkownika umożliwiające wprowadzanie informacji o zamówieniu – nazwij je *Wprowadź zamówienie*. Zadanie *Wprowadź zamówienie* powinno być przypisane użytkownikowi, który uruchamia proces. W ramach wykonania tego zadania powinna być możliwość wpisania numeru ISBN (typu long), imienia (typu string), nazwiska (typu string) zamawiającego oraz uwag (). Wszystkie pola (poza uwagami) powinny być obowiązkowe.

2.3.2. Skrypt w zadaniu *Zatwierdź zamówienie* ustaw na format 'groovy'². Skrypt powinien ustawiać zmienną określającą czy wprowadzony numer ISBN jest poprawny. Na przykład jeżeli chcielibyśmy sprawdzić czy numer mieści się w zakresie 10 – 1000, skrypt mógłby wyglądać następująco:

```
isvalidisbn = (isbn > 10) && (isbn < 1000) ? true : false
```

Powyższa instrukcja tworzy zmienną *isvalidisbn*. Domyślnie zmienna jest lokalna w stosunku do czynności. Zdefiniowanie zmiennej procesu wymaga wywołania funkcji obiektu *execution* (reprezentującego środowisko wykonania – proces) o sygnaturze *setVariable(name, value)*, gdzie parametr *name* jest stringiem nazwy zmiennej globalnej.

2.3.3. Po zadaniu zatwierdzenia dodaj bramkę XOR. Przepływ procesu powinien być kontynuowany do zadania *Opracuj zamówienie* w przypadku gdy zmienna procesu informująca o poprawności numeru ISBN jest ustawiona. W przeciwnym wypadku proces powinien być zatrzymany.

2.3.4. Zadanie *Opracuj zamówienie* powinno być możliwe do wykonania przez użytkowników należących do grupy 'sales'. W ramach tego zadania proces powinien wyświetlać dane wprowadzone na początku procesu (w trybie tylko do odczytu) oraz umożliwiać wpisanie dodatkowej informacji o dacie dostawy.

2.3.5. Zapisz proces, zamknij edytor modelu i umieść proces w przestrzeni silnika (Model action → Deploy).

2.3.6. Przetestuj działanie procesu. Każda jego poprawa wymaga ponownego uruchomienia edytora. Pamiętaj również, że zmieniona definicja procesu musi być ponownie umieszczona w silniku i stanowi nową definicję w stosunku do wersji poprzedniej (sugerowane jest usuwanie wersji poprzednich).

²Język skryptowy platformy Java dostępny domyślnie w Activiti.

Użyteczne linki

1. Specyfikacja BPMN <http://www.omg.org/spec/BPMN/2.0/> (lub lokalnie w folderze z wykładami)
2. Podręcznik użytkownika Activiti: <http://www.activiti.org/userguide/index.html>