

**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCIProjekt współfinansowany przez  
Unię Europejską w ramach  
Europejskiego Funduszu  
Społecznego**UNIA EUROPEJSKA**  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY

|  |                 |  |                           |
|--|-----------------|--|---------------------------|
| <b>Nazwa przedmiotu</b>  |                 | <b>Kod ECTS</b>  |                           |
| Technologie utrwalania dla języka Java   |                 | 11.3.0372  |                           |
| <b>Nazwa jednostki prowadzącej przedmiot</b>   |                 |  |                           |
| Instytut Informatyki   |                 |  |                           |
| <b>Studia</b>  |                 |  |                           |
| <b>wydział</b>   | <b>kierunek</b> | <b>poziom</b>  | <b>pierwszego stopnia</b> |
| Wydział Matematyki,<br>Fizyki i Informatyki  | Informatyka     | <b>forma</b>   | stacjonarne               |
|  |                 | <b>moduł</b>   | wszystkie                 |
|  |                 | <b>specjalnościowy</b>   | wszystkie                 |
|  |                 | <b>specjalizacja</b>   | wszystkie                 |
| <b>Nazwisko osoby prowadzącej (osób prowadzących)</b>  |                 |  |                           |
| dr Jakub Neumann; mgr Tadeusz Puźniakowski; mgr Tadeusz Puźniakowski   |                 |  |                           |
| <b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>  |                 | <b>Liczba punktów ECTS</b>   |                           |
| <b>Formy zajęć</b>   |                 | 3<br>Przedmiot w wymiarze 15h wykładu i 15h laboratorium + praca własna studenta.                    |                           |
| Wykład, Ćw. laboratoryjne  |                 |  |                           |
| <b>Sposób realizacji zajęć</b>   |                 |  |                           |
| zajęcia w sali dydaktycznej  |                 |  |                           |
| <b>Liczba godzin</b>   |                 |  |                           |
| Wykład: 15 godz., Ćw. laboratoryjne: 15 godz.  |                 |  |                           |
| <b>Cykl dydaktyczny</b>  |                 |  |                           |
| 2017/2018 zimowy   |                 |  |                           |
| <b>Status przedmiotu</b>   |                 | <b>Język wykładowy</b>   |                           |
| fakultatywny (do wyboru)   |                 | polski   |                           |
| <b>Metody dydaktyczne</b>  |                 | <b>Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne</b>          |                           |
| <ul style="list-style-type: none"> <li>- Projektowanie doświadczeń</li> <li>- Wykonywanie doświadczeń</li> <li>- Wykład z prezentacją multimedialną</li> <li>- w trakcie wykładu liczne przykłady kodowane (implementowane) na żywo.</li> <li>- ćwiczenia polegają na implementacji fragmentów logiki biznesowej aplikacji wraz z testami, w omawianej aktualnie na wykładzie technologii</li> </ul> |                 | <b>Sposób zaliczenia</b>   |                           |
|  |                 | <ul style="list-style-type: none"> <li>- Zaliczenie na ocenę</li> <li>- Zaliczenie (zal)</li> </ul>  |                           |
|  |                 | <b>Formy zaliczenia</b>  |                           |
|  |                 | ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru |                           |
|  |                 | <b>Podstawowe kryteria oceny</b>   |                           |
|  |                 | Ocenie podlega zrozumienie i umiejętność posługiwania się omawianymi technikami i technologiami.     |                           |
| <b>Sposób weryfikacji założonych efektów kształcenia</b>   |                 |  |                           |
| <b>Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi</b>  |                 |  |                           |
| <b>A. Wymagania formalne</b>   |                 |  |                           |
| Zaliczenie z przedmiotu Technologie Java Enterprise.   |                 |  |                           |
| <b>B. Wymagania wstępne</b>  |                 |  |                           |
| Umiejętność projektowania i programowania obiektowego, dobra znajomość języka Java. Umiejętność posługiwania się programem Maven.  |                 |  |                           |
| <b>Cele kształcenia</b>  |                 |  |                           |
| Celem przedmiotu jest zapoznanie studentów z technologiami i technikami utrwalania danych oraz mapowania obiektowo-relacyjnego dla języka Java wraz z technikami testowania automatycznego.  |                 |  |                           |
| Technologie te pozwalają efektywnie wytworzyć warstwę logiki biznesowej oraz warstwę dostępu do danych (persystencji).   |                 |  |                           |
| <b>Treści programowe</b>   |                 |  |                           |
| Niskopoziomowy dostęp do bazy danych za pomocą JDBC. Koncepty frameworka Spring i jego wsparcie dla technologii persystencji, zarządzanie  |                 |  |                           |

|  |  |
|--|--|
| <p>transakcjami, wsparcie dla testów. Podstawy odwzorowań obiektowo-relacyjnych, Java Persistence API. Framework Hibernate, interfejs HibernateSession, zapytania HQL. Techniki testowania dla omawianych zagadnień.</p>   |  |
| <p><b>Wykaz literatury</b></p> <ul style="list-style-type: none"> <li>• Dokumentacja techniczna dostępna na stronach domowych omawianych technologii</li> <li>• Java Persistence with Hibernate, Christian Bauer, 2006, ISBN-10: 1932394885 lub nowsze wydanie</li> <li>• Hibernate in Action, Christian Bauer, Gavin King, 2004, ISBN: 193239415X lub nowsze wydanie</li> <li>• Spring in Action, Craig Walls, Ryan Breidenbach, 2005, ISBN: 1932394354 lub nowsze</li> </ul> |  |
| <p><b>Efekty kształcenia (obszarowe i kierunkowe)</b></p> <p>Student będzie potrafił wyworzyć warstwę prostej logiki biznesowej (operacje typu CRUD) i dostępu do danych, przy użyciu wybranych technologii (w zależności od potrzeb i w różnych konfiguracjach): JDBC, Hibernate, Spring. Student będzie potrafił pisać własne testy do wytwarzanego kodu i wykonywać je w sposób automatyczny.</p>   | <p><b>Wiedza</b></p> <p>Rozumienie wybranych zagadnień utrwalania danych w programach obiektowych na przykładzie języka Java. Znajomość wybranych aspektów technologii lub API:</p> <ul style="list-style-type: none"> <li>• JDBC</li> <li>• Spring</li> <li>• Hibernate</li> <li>• JUnit</li> </ul>                           |
|  | <p><b>Umiejętności</b></p> <ol style="list-style-type: none"> <li>1. Implementowania niskopoziomowego dostępu do relacyjnych baz danych</li> <li>2. Mapowania obiektowo-relacyjnego</li> <li>3. Automatycznego testowania wytwarzanych fragmentów aplikacji</li> <li>4. Doboru architektury aplikacji i technologii</li> </ol> |
|  | <p><b>Kompetencje społeczne (postawy)</b></p>  |
| <p><b>Kontakt</b></p> <p>kuba@inf.ug.edu.pl</p>  |  |