



Nazwa modułu: Interfejs użytkownika

Rok akademicki: 2017/2018 Kod: JIS-2-010-SW-s Punkty ECTS: 4

Wydział: Fizyki i Informatyki Stosowanej

Kierunek: Informatyka Stosowana Specjalność: Systemy wbudowane i rekonfigurowalne

Poziom studiów: Studia II stopnia Forma i tryb studiów: Stacjonarne

Język wykładowy: Polski Profil kształcenia: Ogólnoakademicki (A) Semestr: 0

Strona www: <http://newton.fis.agh.edu.pl/~wołoszyn/gui/>

Osoba odpowiedzialna: dr hab. inż. Wołoszyn Maciej (wołoszyn@newton.fis.agh.edu.pl)

Osoby prowadzące: dr hab. inż. Wołoszyn Maciej (wołoszyn@newton.fis.agh.edu.pl)  
dr inż. Kawecka-Magiera Barbara (Barbara.Kawecka@fis.agh.edu.pl)

### Krótką charakterystyka modułu

Celem kursu jest praktyczne zapoznanie się z technikami projektowania i programowania graficznego interfejsu użytkownika na różnych platformach systemowych.

### Opis efektów kształcenia dla modułu zajęć

Kod EKM	Student, który zaliczył moduł zajęć wie/umie/potrafi	Powiązania z EKK	Sposób weryfikacji efektów kształcenia (forma zaliczeń)
Wiedza			
M_W001	Student zna zasady projektowania aplikacji zawierających graficzny interfejs użytkownika oraz mechanizmy generowania zdarzeń i sposoby ich obsługi.	IS2A_W15	Aktywność na zajęciach, Udział w dyskusji, Wykonanie ćwiczeń laboratoryjnych
M_W002	Student zna wybrane biblioteki do tworzenia graficznego interfejsu użytkownika dostępne dla języków C, C++, Java i wie jakie są ich wady i zalety w typowych zastosowaniach.	IS2A_W15	Aktywność na zajęciach, Udział w dyskusji, Wykonanie ćwiczeń laboratoryjnych
Umiejętności			
M_U001	Student potrafi tworzyć aplikacje korzystające z graficznego interfejsu użytkownika za pomocą bibliotek takich jak np. Qt, wxWidgets, JFC/Swing.	IS2A_U11	Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych

M_U002	Student potrafi wykorzystać dodatkowe możliwości użytych bibliotek do przygotowania programów w postaci łatwej do kompilacji/uruchamiania na różnych platformach.	IS2A_U17	Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych
M_U003	Student umie samodzielnie wyszukiwać szczegółowe informacje w dostępnej dokumentacji.	IS2A_U01	Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych
M_U004	Student potrafi używać środowisk programistycznych oraz narzędzi wspomagających projektowanie graficznego interfejsu użytkownika.	IS2A_U11	Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych
Kompetencje społeczne			
M_K001	Student posiada umiejętność opisanego własnego rozwiązania w komunikatywny sposób i określenia możliwości jego zastosowania.	IS2A_K06, IS2A_K02	Sprawozdanie, Udział w dyskusji, Wykonanie projektu

## Matryca efektów kształcenia w odniesieniu do form zajęć

Kod EKM	Student, który zaliczył moduł zajęć wie/umie/potrafi	Forma zajęć										
		Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Inne	E-learning
Wiedza												
M_W001	Student zna zasady projektowania aplikacji zawierających graficzny interfejs użytkownika oraz mechanizmy generowania zdarzeń i sposoby ich obsługi.	+	-	+	+	-	-	-	-	-	-	-
M_W002	Student zna wybrane biblioteki do tworzenia graficznego interfejsu użytkownika dostępne dla języków C, C++, Java i wie jakie są ich wady i zalety w typowych zastosowaniach.	+	-	+	+	-	-	-	-	-	-	-
Umiejętności												
M_U001	Student potrafi tworzyć aplikacje korzystające z graficznego interfejsu użytkownika za pomocą bibliotek takich jak np. Qt, wxWidgets, JFC/Swing.	+	-	+	+	-	-	-	-	-	-	-

M_U002	Student potrafi wykorzystać dodatkowe możliwości użytych bibliotek do przygotowania programów w postaci łatwej do kompilacji/uruchamiania na różnych platformach.	+	-	+	+	-	-	-	-	-	-	-
M_U003	Student umie samodzielnie wyszukiwać szczegółowe informacje w dostępnej dokumentacji.	+	-	+	+	-	-	-	-	-	-	-
M_U004	Student potrafi używać środowisk programistycznych oraz narzędzi wspomagających projektowanie graficznego interfejsu użytkownika.	+	-	+	+	-	-	-	-	-	-	-
Kompetencje społeczne												
M_K001	Student posiada umiejętność opisanego własnego rozwiązania w komunikatywny sposób i określenia możliwości jego zastosowania.	-	-	+	+	-	-	-	-	-	-	-

## Treść modułu zajęć (program wykładów i pozostałych zajęć)

### Wykład

#### 1 Wprowadzenie

Znaczenie i podstawowe cechy graficznych interfejsów użytkownika. Przegląd bibliotek i narzędzi.

#### 2 Projektowanie programów z graficznym interfejsem użytkownika

Wzorzec Model-Widok-Kontroler. Zasady projektowania interfejsów użytkownika.

#### 3 Biblioteka Qt

Mechanizm sygnałów i slotów. Narzędzie programistyczne dla Qt na różnych platformach systemowych. Dodatkowe moduły biblioteki Qt.

#### 4 Biblioteka wxWidgets

Obsługa zdarzeń za pomocą biblioteki wxWidgets. Środowiska do projektowania GUI dla wxWidgets.

#### 5 Graficzny interfejs użytkownika w języku Java

Java AWT. JFC/Swing. QtJambi.

### Ćwiczenia laboratoryjne

#### 1 Biblioteka Qt – wstęp

Efekty kształcenia:

- student potrafi utworzyć projekt korzystający z Qt oraz użyć narzędzia qmake do przygotowanie pliku Makefile
- student potrafi skompilować i uruchomić prosty, przykładowy program używający biblioteki Qt
- student potrafi korzystać z dokumentacji biblioteki

#### 2 Tworzenie prostych aplikacji opartych na klasach QWidget i QDialog

Efekty kształcenia:

- student potrafi przygotować program używający własnego okna

- student potrafi wykorzystać klasy biblioteki Qt do tworzenia typowych składników okna (widgetów)

3 Korzystanie z mechanizmu sygnałów i slotów biblioteki Qt

Efekty kształcenia:

- student potrafi zdefiniować własne sygnały

- student potrafi zdefiniować własne sloty

- student potrafi wykorzystać własne i biblioteczne sygnały i sloty do obsługi zdarzeń generowanych przez interfejs użytkownika

4 Tworzenie aplikacji opartych na QMainWindow

Efekty kształcenia:

- student potrafi napisać program posługujący się typowym obiektem klasy dziedziczącej po QMainWindow korzystając ze środowisk programistycznych takich jak np. QtCreator

- student potrafi tworzyć i obsługiwać z poziomu własnej aplikacji paski narzędzi, paski statusu, menu, skróty klawiaturowe

5 Tworzenie aplikacji wykorzystujących JFC/Swing w języku Java

Efekty kształcenia:

- student potrafi przygotować prostą aplikację w języku Java używającą klas Swing-a do tworzenia interfejsu użytkownika

- student potrafi zrealizować obsługę zdarzeń w ramach JFC/Swing

6 Tworzenie graficznego interfejsu użytkownika do aplikacji uruchamianych z linii poleceń oraz uruchamiających wiele wątków

Efekty kształcenia:

- student potrafi posługiwać się klasami używanej biblioteki do uruchamiania wątków i niezależnych procesów

- student potrafi komunikować się z procesami, w tym przysyłać/odbierać dane na/z standardowe strumienie wejściowe/wyjściowe

7 Definiowanie własnych kontrolki (widgetów)

Efekty kształcenia:

- student potrafi przygotować własną klasę definiującą nowy widget

- student potrafi w takiej klasie przygotować metody obsługujące zdarzenia pochodzące z klawiatury, myszy itp.

- student potrafi używać podstawowych operacji graficznych do uzyskiwania zadanego wyglądu własnego widgetu

8 Wykorzystanie wzorca Model-Widok-Kontroler

Efekty kształcenia:

- student potrafi przygotować program wykorzystujący wzorzec Model-Widok-Kontroler np. w ramach mechanizmów udostępnianych przez bibliotekę Qt

### **Ćwiczenia projektowe**

Samodzielnie przygotowanie dwóch aplikacji realizujących to samo zadanie z wykorzystaniem dwóch różnych bibliotek, dowolnie wybranych przez studenta spośród technologii omawianych na wykładzie lub innych uzgodnionych z prowadzącym zajęcia.

### **Sposób obliczania oceny końcowej**

$$K = 0.45 L + 0.55 P$$

gdzie:

L - punkty (wynik procentowy) uzyskane w ramach realizacji ćwiczeń na laboratorium,

P - punkty (wynik procentowy) uzyskane za samodzielnie przygotowany projekt,

K - ocena końcowa (wynik procentowy zostanie przeliczony na ocenę zgodnie z Regulaminem Studiów AGH, przy czym otrzymanie pozytywnej oceny końcowej następuje po uzyskaniu pozytywnych ocen

końcowych z laboratorium oraz projektu).

### **Wymagania wstępne i dodatkowe**

Umiejętność programowania w językach C++ i JAVA

Znajomość podstaw metodologii obiektowej

### **Zalecana literatura i pomoce naukowe**

Ganczarski J., Owczarek M., C++. Wykorzystaj potęgę aplikacji graficznych

Blanchette J., Summerfield M., C++ GUI. Programming with Qt

Ezust A., Ezust P., An Introduction to Design Patterns in C++ with Qt 4

Smart J., Hock K., Csomor S., Cross-Platform GUI Programming with wxWidgets

Schildt H., Java. Kompendium programisty

Wołoszyn M., Programowanie GUI. Notatki do wykładu dostępne na stronie przedmiotu:

<http://newton.fis.agh.edu.pl/~wołoszyn/gui/>

The Java Tutorials, dostępne na stronie <http://docs.oracle.com/javase/tutorial/>

### **Publikacje naukowe osób prowadzących zajęcia związane z tematyką modułu**

- - -

### **Informacje dodatkowe**

- Sposób i tryb wyrównania zaległości powstałych wskutek nieobecności studenta na zajęciach:

Ćwiczenia laboratoryjne i projektowe: Nieobecność na nie więcej niż dwóch zajęciach wymaga od studenta samodzielnego (z możliwością wykorzystania godzin konsultacji) opanowania przerabianego na tych zajęciach materiału. Nieusprawiedliwiona nieobecność na więcej niż dwóch zajęciach oznacza brak możliwości zaliczenia ćwiczeń. Student ma prawo do odrobienia każdej usprawiedliwionej nieobecności na ćwiczeniach laboratoryjnych w wyznaczonym przez prowadzącego terminie, lecz nie później jak w ostatnim tygodniu trwania zajęć.

Obecność na wykładzie: zgodnie z Regulaminem Studiów AGH.

- Zasady zaliczania zajęć:

Ćwiczenia laboratoryjne: Jedynym terminem uzyskania zaliczenia jest koniec zajęć w danym semestrze, przy czym przy obliczaniu końcowej liczby punktów nie będzie brane pod uwagę ćwiczenie, z którego dana osoba uzyskała najmniejszą liczbę punktów. Student, który bez usprawiedliwienia opuścił więcej niż dwa zajęcia nie uzyskuje zaliczenia ćwiczeń; od takiej decyzji prowadzącego zajęcia student może się odwołać do prowadzącego przedmiot (moduł) lub Dziekana.

Ćwiczenia projektowe: Podstawowym terminem uzyskania zaliczenia jest koniec zajęć w danym semestrze. Uzyskanie zaliczenia będzie również możliwe gdy projekt zostanie oddany do końca określonego przez Dziekana terminu uzyskiwania zaliczeń w danym semestrze. Jednym z kryteriów oceny będzie terminowość oddania obu aplikacji składających się na projekt.

- Szczegółowe informacje organizacyjne dotyczące trybu zaliczania zajęć i odrabiania zaległości dostępne są na stronie <http://newton.fis.agh.edu.pl/~wołoszyn/gui/>

**Nakład pracy studenta (bilans punktów ECTS)**

Forma aktywności studenta	Obciążenie studenta
Udział w wykładach	30 godz
Samodzielne studiowanie tematyki zajęć	11 godz
Udział w ćwiczeniach laboratoryjnych	15 godz
Przygotowanie do zajęć	20 godz
Wykonanie projektu	25 godz
Przygotowanie sprawozdania, pracy pisemnej, prezentacji, itp.	4 godz
Udział w ćwiczeniach projektowych	15 godz
Sumaryczne obciążenie pracą studenta	120 godz
Punkty ECTS za moduł	4 ECTS