

## SYLABUS PRZEDMIOTU

1.	Nazwa przedmiotu w języku polskim oraz języku angielskim Indywidualny projekt programistyczny Individual programming project
2.	Dyscyplina naukowa informatyka techniczna i telekomunikacja
3.	Język wykładowy polski
4.	Jednostka prowadząca przedmiot Wydział Fizyki i Astronomii
5.	Rodzaj przedmiotu ( <i>obowiązkowy, do wyboru</i> ) do wyboru
6.	Kierunek studiów Informatyka stosowana i systemy pomiarowe
7.	Poziom studiów I stopień
8.	Rok studiów II rok
9.	Semestr ( <i>zimowy lub letni</i> ) letni
10.	Forma zajęć i liczba godzin laboratorium komputerowe – 30 godzin
11.	Wymagania wstępne w zakresie wiedzy, umiejętności i kompetencji społecznych dla przedmiotu  Podstawy tworzenia aplikacji z interfejsem użytkownika w dowolnej technologii, przy użyciu dowolnego języka i dowolnej biblioteki/frameworka, np. html, python (django, pygame), kotlin, C#, JavaScript, C++ (QT, QtQuick), Swift (język do urządzeń Apple), Unity3D.
12.	Cele kształcenia dla przedmiotu  Uświadomienie studentom istotności planowania projektu (w tym odpowiedniego wyważenia projektu oraz jego elastyczności). Utrwalenie indywidualnych nawyków przy pracy z projektem (systematyczność pracy, regularne debugowanie, testowanie programu na różnych etapach realizacji). Podniesienie poziomu umiejętności dobrego planowania projektu (w tym podział na zadania i podzadania).

	Przygotowanie do pracy w zespole poprzez korzystanie z odpowiedniego systemu kontroli wersji (git). Poprawienie terminowości podczas wykonywania pracy z projektem.	
13.	<p>Treści programowe</p> <p>Studenci wykorzystują posiadane umiejętności i realizują, indywidualne projekty. W czasie zajęć samodzielnie poznają i wykorzystują wybrane przez siebie API.</p> <p>Projektowanie aplikacji i planowanie pracy nad projektem. Systemy wersjonowania i ich użycie w praktyce. Przegląd wybranych bibliotek GUI. Prototypowanie. Prezentacja prototypów projektu.</p>	
14.	<p>Zakładane efekty uczenia się</p> <p>Student:</p> <ul style="list-style-type: none"> <li>– rozumie jak działa proces powstawania oprogramowania;</li> <li>– zna systemy wersjonowania i ich sposoby użytkowania w praktyce;</li> <li>– zna wybrane biblioteki graficznego interfejsu użytkownika (GUI);</li> <li>– rozumie istotę i rolę prototypu w cyklu tworzenia oprogramowania;</li> <li>– potrafi zaprojektować, zaplanować, wykonać i ukończy prosty projekt programistyczny;</li> <li>– potrafi dokonywać zmian w projekcie w celu sprawniejszej jego realizacji;</li> <li>– potrafi zachęcić odbiorcę do użytkowania aplikacji poprzez wykonaną prezentację;</li> <li>– potrafi sprawnie „posługiwać się” swoim kodem oraz wprowadzać niewielkie zmiany zasugerowane przez prowadzącego;</li> <li>– potrafi przedstawić korzyści i trudności wynikające z użycia danej technologii;</li> <li>– potrafi dzielić się swoją wiedzą z innymi i prowadzić dyskusję na temat zaproponowanych w projekcie rozwiązań;</li> <li>– potrafi przedstawić swoje doświadczenia z pracy nad projektem, by udzielić innym wskazówek.</li> </ul>	<p>Symbole odpowiednich kierunkowych efektów uczenia się</p> <p>I1_W04,</p> <p>I1_U06, I1_U08, I1_U16, I1_U17,</p> <p>I1_K02, I1_K04, I1_K05,</p>
15.	<p>Literatura obowiązkowa i zalecana</p> <p>Dokumentacja wybranej technologii dostępna w Internecie Instrukcja i tutorial użytkowania serwisu GitHub dostępne na stronie internetowej</p>	
16.	<p>Metody weryfikacji zakładanych efektów uczenia się:</p> <p>weryfikacja i akceptacja przez prowadzącego założeń projektu przedstawionych w formie pisemnej (przygotowanie opisu propozycji realizowanego projektu w atrakcyjnej formie zachęcającej do zainwestowania w projekt, zawarcie istotnych z punktu widzenia realizacji projektu informacji (m. in. wykorzystanie,</p>	

	<p>opis i uzasadnienie wykorzystanych technologii, podział zadań, harmonogram, koncepcyjne schematy GUI, informacje o repozytorium).</p> <p>przygotowanie i zrealizowanie projektu</p> <p>przygotowanie raportów z realizacji projektu i postępu prac (minimum 2)</p> <p>przygotowanie wystąpienia ustnego (na początku i na zakończenie) na temat realizowanego projektu, dyskusja na temat przyjętych założeń i rozwiązań, udział w dyskusji na temat projektów innych uczestników kursu</p>	
17.	<p>Warunki i forma zaliczenia poszczególnych komponentów przedmiotu:</p> <p>przygotowanie planu projektu</p> <p>wystąpienie ustne prezentujące pomysły i plan projektu</p> <p>przygotowanie i zrealizowanie projektu zgodnego z zaakceptowanym planem projektu</p> <p>wystąpienie ustne, w którym zostanie przedstawiony działający program</p> <p>napisanie raportów z zajęć</p>	
	Nakład pracy studenta wyrażony w godzinach zajęć oraz punktach ECTS	liczba godzin przeznaczona na zrealizowanie danego rodzaju zajęć
	zajęcia (wg planu studiów) z prowadzącym:	
	–laboratorium:	30
	praca własna studenta (w tym udział w pracach grupowych):	
	–przygotowanie do zajęć:	20
	–czytanie wskazanej literatury:	20
	–przygotowanie projektu/wystąpień:	40
	–napisanie raportów z zajęć:	15
	Łączna liczba godzin zajęć	125
	Liczba punktów ECTS ( <i>jeśli jest wymagana</i> )	5