

<b>Kod przedmiotu</b>	02 45 6105 00
<b>Liczba punktów ECTS</b>	3
<b>Nazwa w języku prowadzenia</b>	Zaawansowane aplikacje mobilne
<b>Nazwa w języku polskim</b>	Zaawansowane aplikacje mobilne
<b>Nazwa w języku angielskim</b>	Advanced Mobile Applications
<b>Język prowadzenia zajęć</b>	polski

**Formy zajęć  
Liczba godzin w semestrze**

	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	E-learning
Godziny kontaktowe			30			
Kształcenie na odległość	Nie	Nie	Nie	Nie	Nie	Nie
Udział wagowy w ocenie końcowej.			1,00			

**Jednostka prowadząca**

Katedra Mikroelektroniki i Technik Informatycznych

**Kierownik przedmiotu**

dr inż. Przemysław Sękalski

**Realizatorzy przedmiotu**

dr inż. Paweł Marciniak, dr inż. Przemysław Sękalski, Sebastian Woźniak

**Wymagania wstępne**

Znajomość programowania aplikacji natywnych w systemie Android.  
Umiejętność wykorzystania komunikacji REST.

**Przedmiotowe efekty uczenia się**

1. Student potrafi zaprojektować architekturę aplikacji klient serwer
2. Student potrafi dobrać technologię i narzędzia do realizacji aplikacji klient-serwer
3. Student potrafi pracować w grupie z wykorzystaniem narzędzi pracy współbieżnej
4. Student zna narzędzia ułatwiające pracę współbieżną
5. Student poprawnie wykorzystuje metodykę zwinną w procesie budowania aplikacji
6. Student rozumie cały proces wytwarzania oprogramowania od projektu po testowanie rozwiązania
7. Student poprawnie implementuje funkcjonalności w języku programowania dedykowanym dla urządzeń mobilnych oraz dla serwerów

**Metody weryfikacji przedmiotowych efektów uczenia się**

Dokumentacja projektu prowadzona z wykorzystaniem narzędzi do programowania współbieżnego Raporty z realizacji poszczególnych sprintów w technologii zwinnych Testy wydajnościowe i jednostkowe finalnego rozwiązania

**Kierunkowe**

1. W zaawansowanym stopniu zna i rozumie teoretyczne fundamenty informatyki

**efekty uczenia się**

- oraz wybrane zagadnienia z różnych działów informatyki, zna i rozumie budowę i działanie sprzętu komputerowego oraz podstawowe procesy zachodzące w cyklu życia obiektów i systemów technicznych.
2. Potrafi przeanalizować złożony problem i zaproponować jego rozwiązania, wykazując się kreatywnością w łączeniu wiedzy z zakresu informatyki i innych, specyficznych dla danego problemu, dyscyplin, tworząc specyfikację wymagań dla potrzeb implementacji.
  3. Potrafi łączyć teorie informatyczne, zasady inżynierii oprogramowania oraz wiedzę specjalistyczną z różnych działów informatyki (w procesie budowy rozwiązania problemu).
  4. Potrafi komunikować się, w tym w języku obcym, pracować indywidualnie i w grupie oraz planować i realizować samokształcenie.

**Formy i warunki zaliczenia przedmiotu**

Opracowanie koncepcji aplikacji klient-serwer Przygotowanie listy Product Backlog (story) Finalna ocena zależy od realizacji % wstępnej listy Product Backlog

**Szczegółowe treści przedmiotu**

Sprint 0 Opracowanie koncepcji aplikacji lub omówienie w szczegółach otrzymanego projektu Sprint 1: - Postawienie serwera aplikacyjnego z REST API - Szkic designu aplikacji (Front-end) - np. w programie balsamiq (balsamiq.com) lub dowolnej innej do mock-upów (wszystkie ekrany aplikacji z przejściami) - Wstępna komunikacja klient - serwer z gotowym demonstratorem w postaci np. Ekranu logowania - Dokładny schemat bazy danych (diagram) - Założenie repozytorium grupowe - Conajmniej jeden widok w aplikacji mobilnej z działającym demonstratorem - Możliwość edytowania danych na serwerze z poziomu PC - Opracowanie product backlog ze szczegółami + dodanie na Trello (lub innego narzędzia) - Przynajmniej jedna dodatkowa funkcjonalność z Product backlog Sprint 2: - Implementacja bazy danych (zgodnie z zaprojektowanym w poprzednim sprincie diagramem) - Przetestowanie poprawności działania bazy danych z poziomu aplikacji mobilnej - Możliwość rejestracji nowego użytkownika (zarówno po stronie serwera jak i bazy danych) z poziomu aplikacji mobilnej - Co najmniej dwa widoki aplikacji w tym jeden z funkcjonalnością z Product backlog - Elementy graficzne w aplikacji mobilnej + zastosowanie list / hamburger menu etc. - Przynajmniej jeden unit test sprawdzający wybraną funkcjonalność z Product backlog - 35% Tasków z Product backlog Sprint 3 - 70% Tasków z Product backlog Sprint 4 - 100% Tasków z Product backlog

**Literatura podstawowa**

1. microsoft.learn ([https://docs.microsoft.com/en-gb/learn/?WT.mc\\_id=home\\_homepage-azureportal-learn](https://docs.microsoft.com/en-gb/learn/?WT.mc_id=home_homepage-azureportal-learn))
2. Mobile Apps Engineering: Design, Development, Security, and Testing, Ghita K. Mostefaoui, Faisal Tariq, CRC Press, 7 gru 2018 - 144
3. Microsoft Azure, planning, Deploying and Managing Your Data Center in the Cloud, Copeland, 2015
4. Optymalizacja wydajności aplikacji na Android, Herv Guihot, Helion, 2 lip 2013 - 256
5. <https://docs.aws.amazon.com/>
6. <https://cloud.google.com/docs>
7. <https://flutter.dev/docs>
8. <https://developer.android.com/docs>

**Literatura uzupełniająca**

1. strony internetowe: stackoverflow, github, flutter, android, go i inne

**Bilans godzin**

Forma zajęć	Liczba godzin
Laboratorium	30
Praca samodzielna i w zespole	50
Udział w konsultacjach	5
Udział w pisemnych i/lub praktycznych formach weryfikacji	5

Forma zajęć	Liczba godzin
SUMA :	90

***Uwagi***

***Data aktualizacja karty*** 2021-11-08 09:49:18