

Instrukcja do laboratorium z przedmiotów:

- Projektowanie aplikacji mobilnych
- Systemy operacyjne na platformach mobilnych
- Przetwarzanie w chmurze dla mobilnych aplikacji

Warunkiem zaliczenia laboratorium jest oddanie następujących programów:

1. Kalkulator
2. AstroWeather 1 (Astro)
3. AstroWeather 2 (Pogoda)

Terminy oddawania programów:

Numer zjazdu	3	4	5	6	7	8	9	10	+
-	Kalkulator 5	Kalkulator 4	Kalkulator 3	Astro 5	Astro 4	Astro 3	Pogoda 5	Pogoda 4	Pogoda 3

W przypadku oddania programu po pierwszym terminie, każdy kolejny termin obniża ocenę o 1 (Oddanie programu kalkulator na zajeździe numer 3 daje możliwość uzyskania maksymalnie oceny dobrej (4) etc.).

Ocena z laboratorium będzie średnią arytmetyczną ocen uzyskanych z każdego oddawanego programu.

Oddanie programu po terminie na ocenę 3 skutkuje możliwością uzyskania maksymalnie oceny 3 jako finalnej z laboratorium niezależnie od uzyskanych ocen z innych programów (na przykład uzyskanie oceny 5 z kalkulatora, oddanie astro po terminie na ocenę 3 i oddanie pogody na ocenę 5 da finalną ocenę 3).

Program musi zostać napisany na platformę Android w **Android Studio** (nie ma możliwości korzystania z innych środowisk) w języku **Java lub Kotlin** (nie ma możliwości pisania w języku Cordova, Xamarin etc.).

Program musi pracować na najnowszych dostępnych wersjach systemu Androida oraz urządzeniach znajdujących się a laboratorium tj:

- Telefon Samsung Galaxy S5 (Android 6.0 - API level 23)
- Tablet Samsung Galaxy Note 10.1 (Android 5.1.1 - API level 22)

Na pierwszych zajęciach należy sprawdzić jaką aktualnie wersję oprogramowania posiadają wspomniane urządzenia i dostosować do nich swoje przyszłe programy.

Programy oddawane na zajęciach będą testowane na w/w. urządzeniach, zarówno w orientacji pionowej jak i poziomej.

Istnieje możliwość pracy na własnych laptopach.

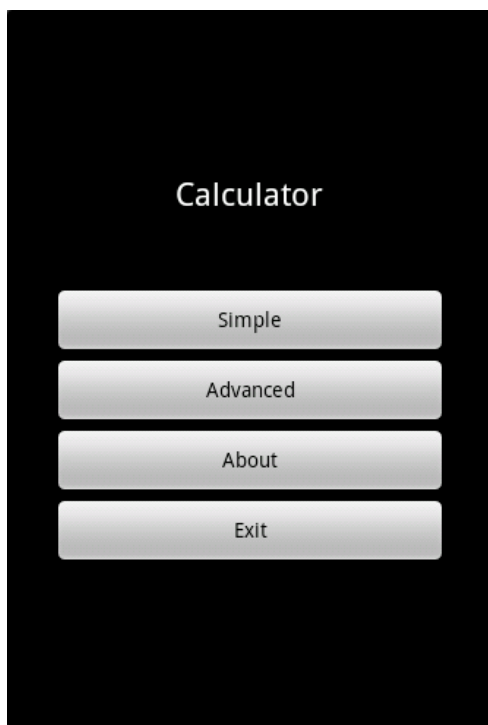
Ćwiczenie 1: Kalkulator

Celem ćwiczenia jest zapoznanie się z podstawowymi elementami interfejsu użytkownika (layouty, przyciski, pola tekstowe) oraz sposobem tworzenia aktywności i przełączania między nimi.

Założenia:

- 1) Aplikacja powinna zapewniać funkcjonalność kalkulatora, który w zależności od wyboru użytkownika pozwoli na wykonanie podstawowych lub zaawansowanych obliczeń.
- 2) Ekran startowy aplikacji powinien umożliwiać:
 - Wybór typu kalkulatora – prosty lub zaawansowany.
 - Wyświetlenie informacji o aplikacji (np. nazwa aplikacji, nazwisko autora, krótki opis).
 - Zamknięcie aplikacji.
- 3) Kalkulator w wersji prostej powinien umożliwiać wykonanie jedynie podstawowych operacji matematycznych (zmiennoprzecinkowych) w tym obowiązkowo:
 - Cztery podstawowe działania: dodawanie (+), odejmowanie(-), mnożenie(*), dzielenie(/).
 - Zmiana znaku liczby (+/-) - znak zmienia się po, a nie przed wprowadzeniem liczby.
 - Clear enter/clear (C/CE) - pierwsze naciśnięcie tego klawisza kasuje liczbę na wskaźniku i umożliwia wprowadzenie innej, bez zakłócania toku obliczeń. Dwukrotne naciśnięcie zeruje wszystkie rejestry kalkulatora (chyba, że istnieje zaimplementowana funkcja pamięci).
 - All clear (AC) - klawisz całkowitego kasowania rejestrów (i opcjonalnie pamięci) kalkulatora.
 - Wyświetlanie wyniku działania (=).
- 4) Kalkulator w wersji zaawansowanej, oprócz podstawowych, powinien dodatkowo implementować bardziej zaawansowane operacje w tym obowiązkowo:
 - Funkcje trygonometryczne: sinus (sin), cosinus (cos), tangens (tan).
 - Funkcje logarytmiczne: logarytm naturalny (ln), logarytm (log).
 - Obliczanie procentów (%).
 - Pierwiastkowanie (sqrt).
 - Potęgowanie (x^2), (x^y).

- 5) Dane wprowadzane przez użytkownika powinny być weryfikowane. W przypadku błędnych danych użytkownik powinien zostać o tym poinformowany (można w tym celu wykorzystać np. powiadomienie Toast).
- 6) Kalkulator musi pracować poprawnie w orientacji pionowej i poziomej. Operacja wprowadzona w trybie pionowym, może zostać kontynuowana z trybie poziomym.
- 7) Przykładowe zrzuty ekranów:



Ćwiczenie 2: AstroWeather część 1

Celem ćwiczenia jest zapoznanie się z następującymi zagadnieniami:

- Interfejs użytkownika bazujący na fragmentach.
- Komunikacja pomiędzy fragmentami i aktywnością.
- Dostosowanie interfejsu użytkownika do różnych ekranów.

Założenia:

- 1) Zadaniem aplikacji jest obliczanie i prezentacja informacji astronomicznych dotyczących Słońca i Księżyca.
- 2) Informacje prezentowane przez aplikację to:
 - A. Dla Słońca:
 - Wschód (czas i azymut).
 - Zachód (czas i azymut).
 - Zmierz i świt cywilny (czas).
 - B. Dla Księżyca:
 - Wschód i zachód (czas).
 - Najbliższy nów i pełnia (data).
 - Faza Księżyca (w procentach).
 - Dzień miesiąca synodycznego.
- 3) Ponadto aplikacja powinna wyświetlać aktualny czas urządzenia (odświeżany co 1 sekundę) oraz lokalizację wprowadzoną przez użytkownika (długość i szerokość geograficzną), dla której obliczane są w/w dane.
- 4) Informacje astronomiczne powinny być odświeżane co pewien czas, zdefiniowany przez użytkownika w ustawieniach (np. co 15 minut).
- 5) Do obliczenia informacji astronomicznych można wykorzystać bibliotekę AstroCalculator. Bibliotekę w formie pliku JAR można znaleźć pod adresem: http://fiona.dmcs.pl/~pperek/pam_2015_2016/AstroCalculator_lib/astrocalculator.jar
Dokumentacja biblioteki znajduje się pod adresem: http://fiona.dmcs.pl/~pperek/pam_2015_2016/AstroCalculator_lib/doc/index.html

- 6) Obliczone dane powinny być prezentowane w przejrzystej formie niezależnie od typu urządzenia (telefon/tablet). W tym celu należy zaprojektować interfejs bazujący na co najmniej dwóch fragmentach (1 – dla Słońca, 2 – dla Księżyca).

Układ fragmentów na ekranie powinien być zależny od jego orientacji i rozdzielczości. Aplikacja powinna uwzględniać co najmniej 4 różne layouty (tablet/telefon, orientacja pionowa/pozioma).

W układzie pionowym na telefonie fragmenty powinny być przewijane (w tym celu można wykorzystać widok ViewPager). W przypadku tabletów wszystkie fragmenty mogą być jednocześnie widoczne na ekranie.

- 7) W aplikacji powinno być dostępne menu, które umożliwi ustawienie:
- Lokalizacji (długość i szerokość geograficzna).
 - Częstotliwości odświeżania danych astronomicznych (można w tym celu wykorzystać Android Spinner adapter).

Ćwiczenie 3: AstroWeather część 2

Celem ćwiczenia jest zapoznanie się z następującymi zagadnieniami:

- Usługi sieciowe dostępne w systemie Android.
- Parsowanie plików XML/JSON.
- Przechowywanie danych w pamięci urządzenia (np. w prywatnym katalogu aplikacji).
- Przechowywanie danych w bazie danych.
- Interfejs użytkownika bazujący na fragmentach.

Założenia:

- 1) Niniejsze ćwiczenie zakłada rozszerzenie aplikacji przygotowanej w ćwiczeniu 2. o dodatkowe funkcjonalności umożliwiające pobieranie i wyświetlanie informacji o warunkach pogodowych (bieżących i prognozowanych) dla lokalizacji wybranych przez użytkownika.
- 2) Dane pogodowe można pobrać korzystając z API udostępnianego przez serwis openweathermap.org. Dane można pobierać w postaci plików JSON lub XML. Szczegółowe informacje dla programistów chcących korzystać z plików udostępnianych w ramach portalu openweathermap.org można znaleźć pod adresem: <https://openweathermap.org/api>. Przykładowe zapytanie na podstawie nazwy miejscowości `api.openweathermap.org/data/2.5/weather?q={cityname}&appid={APIkey}`
- 3) Aplikacja powinna umożliwiać użytkownikowi zdefiniowanie listy ulubionych lokalizacji, dla których będą pobierane dane pogodowe. Po wprowadzeniu lokalizacji aplikacja sprawdza w serwisie openweathermap.org czy wpisane dane są poprawne i pobiera je, a następnie zapamiętuje lokalizację w bazie danych.
- 4) W momencie uruchomienia aplikacja powinna sprawdzać czy możliwe jest połączenie z Internetem. Jeżeli tak, pobierane są aktualne informacje na temat pogody i zapisywane w pamięci telefonu (w prywatnym katalogu aplikacji).
- 5) Jeżeli żadne połączenie internetowe nie jest aktywne w chwili uruchomienia aplikacji, informacje na temat pogody powinny zostać wczytane z pliku, który został zapisany podczas ostatniego połączenia. Ponadto użytkownik powinien być poinformowany o tym, że dane mogą być nieaktualne, a do aktualizacji wymagane jest połączenie internetowe.

Uwaga: Warto zastanowić się czy konieczne jest pobieranie informacji z Internetu przy każdym uruchomieniu aplikacji. Wśród informacji dostarczanych w pliku XML znajduje się czas, przez który dane mogą być trzymane w pamięci podręcznej, bez konieczności odświeżania. Można również z góry założyć pewien czas, przez który dane nie będą odświeżane.

- 6) Dane pogodowe powinny być prezentowane w przejrzystej formie. W tym celu należy rozbudować dotychczasowy interfejs aplikacji z ćwiczenia 2. o dodatkowe fragmenty, np.:
- Fragment 1 – podstawowe dane, tj.: nazwa miejscowości, współrzędne geograficzne, czas, temperatura, ciśnienie, opis i reprezentacja graficzna warunków pogodowych.
 - Fragment 2 – dane dodatkowe np.: informacje o sile i kierunku wiatru, wilgotności, widoczności.
 - Fragment 3 – prognoza pogody na nadchodzące dni.

Układ fragmentów na ekranie powinien być zależny od jego orientacji i rozdzielczości.

- 7) W aplikacji powinno być dostępne menu, które umożliwi:
- Odświeżenie informacji z Internetu na żądanie użytkownika.
 - Ustawienie/zmianę lokalizacji, dla których pobierane są dane.
 - Wybór jednostek miary stosowanych w aplikacji.

Uwaga!

W przypadku, gdy prowadzący będzie miał podejrzenia co do samodzielności napisanej aplikacji, może wyznaczyć jako zadanie indywidualne napisanie alternatywnej aplikacji o podobnych wymaganiach co splotatowana aplikacja. Termin i ocena oddawania alternatywnej aplikacji będą identyczne jak aplikacji podstawowej tj. w przypadku "wychwycenia" plagiatu / niesamodzielnie napisanej aplikacji, prowadzący wyznacza zakres nowej aplikacji. Student, będąc "przyłapanym" na niesamodzielnej pracy w dniu pierwszego terminu oddawania pracy zostanie poproszony o napisanie aplikacji alternatywnej. W kolejnym terminie automatycznie najwyższą oceną jaką będzie mógł otrzymać to ocena dobra (4) - zgodnie z systemem oceniania opisanym na stronie 1.