

Sprawdzian praktyczny – Podstawy Android

Informacje wstępne:

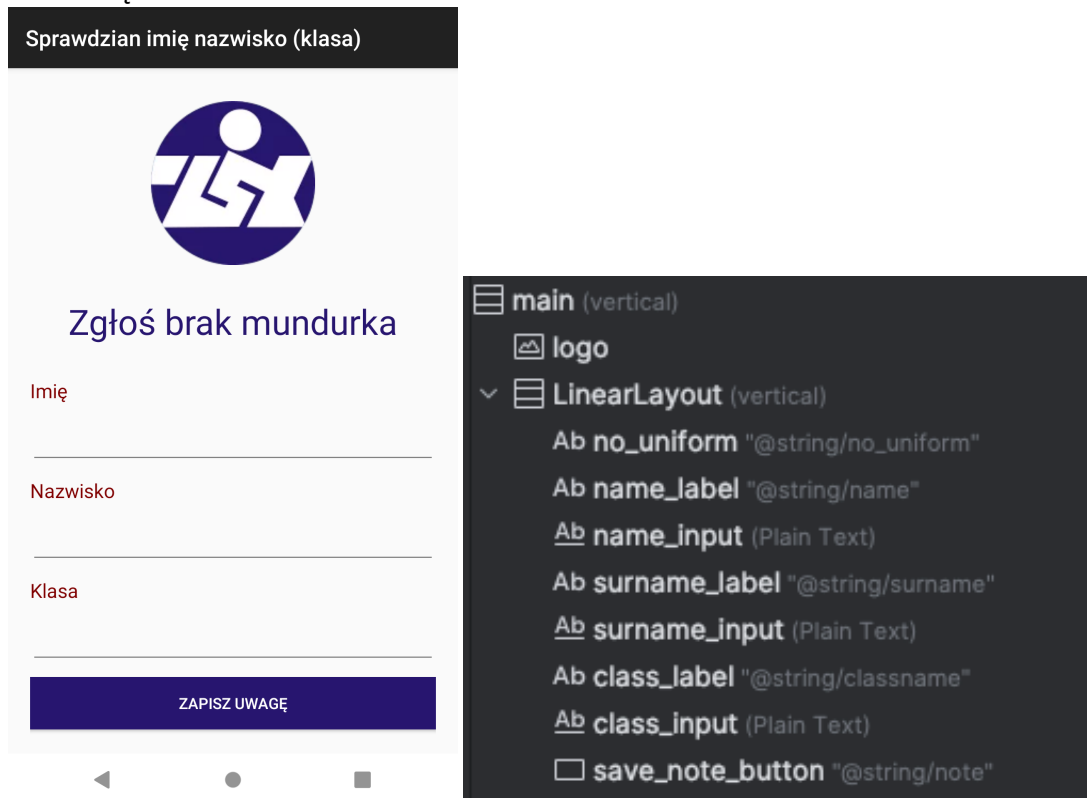
1. W folderze początkowym załączono nagranie ekranu z działającą aplikacją oraz grafikę zawierającą logo ZSK.
2. Podczas sprawdzianu zabronione jest korzystanie z zasobów sieci Internet.
3. Dla sprawdzianu przewidywany jest **jeden termin poprawkowy** (jeżeli będzie konieczny).
4. Wszelkie próby nieuczciwego pisania sprawdzianu **zakończą się anulowaniem podejścia oraz zastosowaniem wyższych progów punktowych przy kolejnym podejściu. Kolejne podejście automatycznie stanie się jedynym możliwym podejściem.**

Powodzenia 😊

Przygotuj aplikację Android zgodnie z poniższymi wymaganiami.

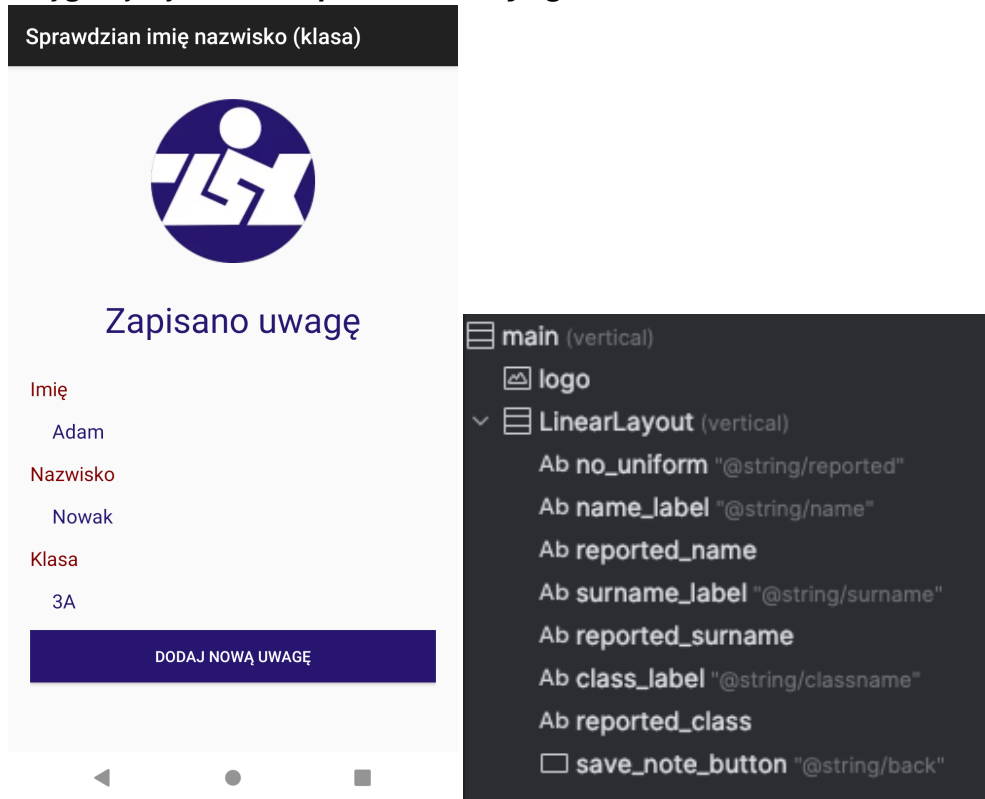
1. Utwórz nowy projekt w Android Studio:
 - a. Template: **Empty Views Activity**,
 - b. Nazwa aplikacji: **Sprawdzian <imię> <nazwisko> <klasa>**,
 - c. Nazwa pakietu: **edu.zsk.imię_nazwisko_klasa**,
 - d. Język programowania: **Java**,
 - e. Minimalne SDK: **API 29**,
 - f. Konfiguracja budowania: **Kotlin DSL**.
2. Usuń motyw dla trybu ciemnego.
3. Ustaw motyw bazowy na **Theme.AppCompat.Light.DarkActionBar**.
4. Dodaj atrybut motywu, który sprawi, że aplikacja będzie wyświetlana w trybie pełnoekranowym.
5. Zdefiniuj kolory:
 - a. **primaryColor** o wartości **#FF27156F**
 - b. **labelColor** o wartości **#FF800000**
6. Utwórz arkusz stylów zawierający następujące reguły:
 - a. reguła **heading**:
 - i. kolor tekstu: **primaryColor**
 - ii. rozmiar tekstu: **32sp**
 - iii. ułożenie tekstu: **do środka**
 - iv. margines dolny: **30dp**
 - b. reguła **label**:
 - i. kolor tekstu: **labelColor**
 - ii. rozmiar tekstu: **18sp**
 - iii. ułożenie tekstu: **do lewej**

- iv. margines dolny: **15dp**
 - c. reguła **reported**:
 - i. dziedziczy po regule **label**
 - ii. kolor tekstu: **primaryColor**
 - iii. margines lewy: **20dp**
 - d. reguła **input**:
 - i. rozmiar tekstu: **16sp**
 - ii. ułożenie tekstu: **do lewej**
 - iii. margines dolny: **10dp**
 - iv. kolor tekstu: **primaryColor**
 - e. reguła **button**:
 - i. kolor tła: **primaryColor**
 - ii. kolor tekstu: **biały**
7. Przygotuj layout dla głównej aktywności zgodnie ze zrzutem ekranu oraz strukturą:



- a. Dla layoutu o identyfikatorze **main** zastosuj następujące dane:
 - i. Typ layoutu: **Linear Layout (vertical)**
 - ii. Szerokość, wysokość: **dopasowanie do rodzica**
- b. Dla obrazka dodaj następujące atrybuty:
 - i. Margines górny: **30dp**
 - ii. Margines dolny: **30dp**
 - iii. Opis: **Logo ZSK**
 - iv. Wysokość: **dopasowanie do zawartości**
 - v. Szerokość: **dopasowanie do rodzica**
- c. Dla zagnieżdżonego LinearLayout zastosuj następujące właściwości:
 - i. Szerokość, wysokość: **dopasowanie do rodzica**
 - ii. Ułożenie layoutu: **wertykalne**

- iii. Margines lewy i prawy: **20dp**
 - d. Dla tekstu „Zgłoś brak mundurka” zastosuj stylowanie **heading**
 - e. Dla tekstów „Imię”, „Nazwisko” oraz „Klasa” zastosuj stylowanie **label**
 - f. Dla elementów typu EditText zastosuj stylowanie **input**
 - g. Dla przycisku zastosuj stylowanie **button**
8. Utwórz drugą aktywność o nazwie **ReportedActivity**
9. Przygotuj layout dla **ReportedActivity** zgodnie ze zrzutem ekranu oraz strukturą:



- a. Dla layoutu o identyfikatorze **main** zastosuj następujące dane:
 - i. Typ layoutu: **Linear Layout (vertical)**
 - ii. Szerokość, wysokość: **dopasowanie do rodzica**
 - b. Dla obrazka dodaj następujące atrybuty:
 - i. Margines górny: **30dp**
 - ii. Margines dolny: **30dp**
 - iii. Opis: **Logo ZSK**
 - iv. Wysokość: **dopasowanie do zawartości**
 - v. Szerokość: **dopasowanie do rodzica**
 - c. Dla zagnieżdżonego LinearLayout zastosuj następujące właściwości:
 - i. Szerokość, wysokość: **dopasowanie do rodzica**
 - ii. Ułożenie layoutu: **wertykalne**
 - iii. Margines lewy i prawy: **20dp**
 - d. Dla tekstu „Zapisano uwagę” zastosuj stylowanie **heading**
 - e. Dla tekstów „Imię”, „Nazwisko” oraz „Klasa” zastosuj stylowanie **label**
 - f. Dla elementów **reported_name**, **reported_surname** oraz **reposted_class** zastosuj stylowanie **reported**
 - g. Dla przycisku zastosuj stylowanie **button**
10. Opracuj logikę aplikacji zgodną z poniższymi wymaganiami:

- a. Po kliknięciu w przycisk **Zapis uwagę** zostaną wykonane następujące akcje:
 - i. W przypadku braku wypełnienia któregośkolwiek z pól zostanie wyświetlony Toast Message o treści „Wypełnij wszystkie pola!”
 - ii. W przypadku wypełnienia wszystkich pól wykonają się następujące akcje:
 - 1. Z pól edycyjnych zostaną pobrane dane
 - 2. Zostanie utworzona nowa intencja, do której dane będą przekazane
 - 3. Zostanie wyświetlone okno dialogowe postępu (**android.app.ProcessDialog**) z tytułem **Dodaję uwagę...** oraz wiadomością **Proszę czekać**. Okno dialogowe będzie widoczne 2 sekundy.
 - 4. W czasie wyświetlania okna dialogowego pola edycyjne zostaną wyczyszczone.
 - 5. Po upływie dwóch sekund okno dialogowe zostanie zamknięte, a użytkownik zostanie przekierowany do aktywności **ReportedActivity**
 - 6. Aktywność **ReportedActivity** uzupełni danymi pola oznaczone identyfikatorami **reported_name**, **reported_surname** oraz **reported_class**.
 - b. Po kliknięciu w przycisk **Dodaj nową uwagę** aktywność **ReportedActivity** zostanie zakończona, a użytkownik wróci do pierwszej aktywności.
11. Wymagania dodatkowe:
- a. Wszystkie teksty powinny być zapisane w pliku **strings.xml**
 - b. Po zakończeniu pracy należy spakować projekt do folderu skompresowanego o nazwie
<klasa>_<nazwisko>_<imię>_sprawdzian_android.zip