

Program szkolenia:

Zaawansowane tworzenie widoków, profilowanie i optymalizacja aplikacji Androidowych

Informacje:

Nazwa:	Zaawansowane tworzenie widoków, profilowanie i optymalizacja aplikacji Androidowych
Kod:	android-android-adv
Kategoria:	Android
Grupa docelowa:	developerzy
Czas trwania:	2 dni
Forma:	40% wykłady / 60% warsztaty

Szkolenie zostało przygotowane z myślą o programistach platformy Android pragnących poszerzyć swój warsztat, poznać nowe podejścia i narzędzia oraz przygotować się do projektów większej skali w celu uniknięcia typowych pułapek.

Materiały wstępne

Przed szkoleniem możesz zapoznać się z serią naszych artykułów: [Zaawansowane programowanie na platformie Android](#).

Zalety szkolenia:

- Praktyczna wiedza zdobyta w projektach dużej skali
- Architektura i wzorce projektowe
- Typowe pułapki oraz najlepsze praktyki

Szczegółowy program:

1. Zaawansowane tworzenie interfejsu użytkownika.

1.1. Tworzenie własnych elementów interfejsu użytkownika

1.1.1. Agregacja wielu widoków w celu wielokrotnego użycia i zastosowania SRP w warstwie widoku

1.1.2. Bezpośrednie rysowanie własnych elementów interfejsu użytkownika

1.1.3. Stylowanie własnych widoków

1.1.4. Obsługa interakcji użytkownika

1.1.5. Zarządzanie stanem w widokach

2. Optymalizacja działania aplikacji Androidowych

2.1. Profilowanie działania aplikacji

2.1.1. Przegląd narzędzi

2.1.2. Monitorowanie wydajności

2.1.3. Strategie znajdowania źródeł problemów z wydajnością

2.2. Optymalizacje warstwy widoku

2.3. Optymalizacje widoków opartych o adaptery (ListView / RecyclerView)

2.4. Wykrywanie i zapobieganie wyciekom pamięci

2.4.1. Najczęstsze przyczyny wycieków pamięci w środowisku Androida

2.4.2. Strategie i narzędzia wspomagające detekcję źródeł wycieków pamięci

2.5. Optymalne ładowanie i zarządzanie bitmapami

2.5.1. Optymalizacja rozmiaru ładowanych bitmap

2.5.2. Strategie bezpiecznego cachowania bitmap do wielokrotnego użycia

2.5.3. Wykorzystanie LRU cache do przechowywania wielu dużych obiektów w pamięci aplikacji

2.6. Wydajne zapisywanie stanu

2.7. Optymalizacja kodu pod kątem maszyny wirtualnej Dalvik/ART

