

Apps Deployment to Playstore

Raih Mimpi #TanpaBatas



Case Study #1

- ☐ Penjelasan Project
- ☐ Case Study

Case Study #1

Saat menggunakan Expo untuk pengembangan aplikasi, proses deployment ke Google Play Store sedikit berbeda dibandingkan dengan aplikasi React Native murni. Expo menyediakan alur kerja yang lebih sederhana dan terotomatisasi untuk membangun (build) dan menerbitkan aplikasi.

Case Study #1

- **Pembuatan Android App Bundle dengan Expo:**

Dengan menggunakan Expo, Anda tidak perlu secara manual membuat Android App Bundle. Expo menyediakan perintah `expo build:android` yang mengotomatisasi proses ini. Perintah ini akan membangun versi rilis dari aplikasi Anda dalam bentuk Android App Bundle atau APK.

Sebelum menjalankan perintah ini, pastikan `app.json` sudah dikonfigurasi dengan benar, termasuk `sdkVersion`, `versionCode`, dan informasi lainnya yang relevan.

Case Study #1

- **Keystore:**

Untuk keamanan, Expo mengelola Keystore secara otomatis untuk Anda. Ketika Anda melakukan build pertama kali, Expo akan membuat Keystore dan menyimpannya di server Expo. Anda juga memiliki opsi untuk mengunggah keystore kustom jika sudah memiliki satu.

Case Study #1

- **Penerbitan ke Google Play Store:**

Setelah build selesai, Expo akan memberikan link untuk mengunduh file .aab (Android App Bundle) atau .apk.

Anda perlu mengunggah file ini ke Google Play Console, mirip dengan proses untuk aplikasi React Native murni.

Isi semua detail yang diperlukan di Google Play Console, seperti deskripsi, screenshot, dan lain-lain.

Kirim aplikasi untuk review dan setelah disetujui, aplikasi Anda akan dipublikasikan di Google Play Store.

Case Study #1

- **Pembaruan Aplikasi:**

Expo juga menyediakan fitur OTA (Over-the-Air) updates, yang memungkinkan Anda mengirim pembaruan langsung ke pengguna tanpa perlu melewati proses review Google Play setiap kali.

Case Study #1

- **Pembuatan Account Developer di Google Play:**

Seperti pada proses standar, Anda tetap memerlukan akun developer Google Play untuk menerbitkan aplikasi.

Pengaturan Continuous Integration (CI)

- **Automasi Tes dan Build:**

Otomatisasi tes unit, integrasi, dan end-to-end setiap kali ada perubahan kode (biasanya pada saat merge request atau push ke branch tertentu).

Otomatisasi proses build untuk menghasilkan artefak yang siap rilis, seperti Android App Bundle atau APK untuk aplikasi Android.

Pengaturan Continuous Integration (CI)

- **Tools CI Populer:**

Gunakan tools seperti Jenkins, CircleCI, Travis CI, atau GitHub Actions untuk mengatur pipeline CI Anda.

Konfigurasikan skrip untuk menjalankan tes dan build otomatis berdasarkan trigger yang telah ditentukan.

Pengaturan Continuous Deployment (CD) dengan Expo

- **Integrasi dengan Expo:**

Jika menggunakan Expo, Anda dapat memanfaatkan Expo CLI dalam pipeline CD Anda untuk mengotomatisasi proses deployment.

Gunakan perintah seperti `expo publish` untuk pembaruan OTA atau `expo build:android` untuk build production.

Pengaturan Continuous Deployment (CD) dengan Expo

- **Otomasi Deployment ke Google Play Store:**

Setelah build selesai, otomatisasi proses unggah ke Google Play Store.

Tools seperti Fastlane dapat sangat membantu dalam mengotomatisasi proses unggah ke Google Play Store, termasuk pengisian changelog, pemilihan track rilis (prod, beta, alpha), dan lainnya.

Manajemen Konfigurasi dan Keamanan

- **Pengelolaan Keystore:**

Dalam konteks Expo, Expo sudah mengelola keystore. Namun, jika Anda melakukan eject dari Expo atau menggunakan React Native murni, penting untuk mengelola keystore dengan aman.

Simpan keystore dalam penyimpanan rahasia yang aman dan akses dalam pipeline CD.

Manajemen Konfigurasi dan Keamanan

- **Variabel Lingkungan dan Konfigurasi Rahasia:**

Gunakan variabel lingkungan untuk mengelola konfigurasi yang berubah-ubah antar lingkungan (development, staging, production).

Pastikan kunci API dan konfigurasi rahasia lainnya tidak termuat dalam kode sumber dan aman dari akses tidak sah.

Monitoring dan Feedback

- **Penggunaan Log dan Monitoring:**

Integrasikan alat pemantauan untuk memantau kesehatan aplikasi setelah deployment.

Gunakan log untuk menganalisis masalah yang terjadi selama proses CI/CD.

Monitoring dan Feedback

- **Feedback Loop:**

Buat sistem untuk secara cepat menerima dan menanggapi feedback dari pengguna atau dari sistem monitoring Anda.

Contoh Workflow GitHub Actions untuk React Native dengan Expo

- **Buat File Workflow di Repository GitHub Anda**

Buat file baru di `.github/workflows/main.yml` di repository Anda.

- **Isi File Workflow**

Berikut adalah contoh isi file `main.yml` untuk menjalankan tes dan build menggunakan Expo:

Contoh Workflow GitHub Actions untuk React Native dengan Expo

```
1 name: React Native CI/CD
2
3 on:
4   push:
5     branches: [ main ]
6   pull_request:
7     branches: [ main ]
8
9 jobs:
10  test-and-build:
11    runs-on: ubuntu-latest
12
13    steps:
14      - uses: actions/checkout@v2
15
16      - name: Set up Node.js
17        uses: actions/setup-node@v2
18        with:
19          node-version: '14'
20
21      - name: Install Dependencies
22        run: npm install
23
24      - name: Run Tests
25        run: npm test
26
27      - name: Install Expo CLI
28        run: npm install -g expo-cli
29
30      - name: Login to Expo
31        run: expo login --non-interactive -u ${ secrets.EXPO_CLI_USERNAME } -p ${ secrets.EXPO_CLI_PASSWORD }
32
33      - name: Publish to Expo
34        run: expo publish
35
36      # Anda dapat menambahkan langkah tambahan untuk build dan deploy
37      # ke Google Play Store menggunakan Fastlane atau alat serupa di sini
38
```

Contoh Workflow GitHub Actions untuk React Native dengan Expo

Dalam contoh ini:

Trigger: Workflow dijalankan pada setiap push dan pull request ke branch main.

Set up Node.js: Menyiapkan lingkungan Node.js.

Install Dependencies: Menginstal dependencies proyek.

Run Tests: Menjalankan tes.

Install Expo CLI dan Login: Menginstal Expo CLI dan login menggunakan credentials yang disimpan di GitHub secrets.

Publish to Expo: Menerbitkan aplikasi ke Expo.

Contoh Workflow GitHub Actions untuk React Native dengan Expo

- **Konfigurasi Secrets di GitHub**

Pergi ke Settings repository GitHub Anda, lalu ke bagian Secrets.

Tambahkan EXPO_CLI_USERNAME dan EXPO_CLI_PASSWORD dengan kredensial Expo CLI Anda.

- **Tambahkan Langkah Tambahan untuk Deployment**

Untuk deployment ke Google Play Store, Anda bisa menggunakan alat seperti Fastlane. Ini memerlukan konfigurasi tambahan dan integrasi dengan Google Play Developer API.



Thank you