

Bezpieczeństwo aplikacji mobilnych

PROJEKT

Menadżer haseł

Jakub Jach
Patryk Jaworski
11D24A

1. Wykorzystane technologie

W ramach projektu wykorzystano następujące technologie:

Część backend – napisana w języku Python w stylu API REST:

- django – framework do tworzenia aplikacji internetowych i webowych w oparciu o protokół HTTP
- django-otp – biblioteka umożliwiająca użycie mechanizmu OTP w ramach aplikacji django
- django-cryptography – pozwala na szyfrowanie danych zapisywanych w modelu ORM dla modeli django
- django-rest-framework – framework umożliwiający tworzenie interfejsów API REST z użyciem django – udostępnia serializery pozwalające na zwracanie obiektów i wybranych pól w postaci JSON
- base64 – umożliwia zakodowanie danych binarnych do odpowiedniej postaci – wykorzystywane do zapisywania tajnego klucza urządzenia do bazy
- environ – biblioteka pozwalająca na definiowanie zmiennych środowiskowych w zewnętrznych plikach co pozwala uniknąć problemów z hardkodowaniem

Właściwa aplikacja mobilna – napisana w języku JavaScript z użyciem Expo:

- react-native – framework do tworzenia aplikacji mobilnych z użyciem języka JavaScript
- react-native-async-storage – pozwala na dostęp do lokalnej pamięci z poziomu JavaScript
- react-native-clipboard – umożliwia dostęp do schowka np. skopiowanie do niego tekstu
- react-native-picker – biblioteka do stworzenia menu z opcjami do wyboru
- react-native-biometrics – pozwala na dostęp do biometrii urządzenia
- react-native-gesture-handler – obsługa gestów
- react-native-reanimated – animacje
- react-navigation/drawer – biblioteka do tworzenia nawigacji typu drawer
- react-navigation/stack – nawigacja typu stack
- axios – biblioteka do komunikacji z serwerem HTTP / HTTPS
- expo-local-authentication – biblioteka do lokalnego logowania, używana w połączeniu z biometrią
- expo-secure-store – biblioteka do obsługi mechanizmów bezpiecznego lokalnego przechowywania danych
- expo-status-bar – biblioteka do obsługi status bar w aplikacji

2. Uruchomienie projektu – backend

W celu uruchomienia części backend projektu konieczne jest zainstalowanie wymienionych bibliotek w systemie. Aby uniknąć konfliktów z już zainstalowanymi bibliotekami zalecane jest stworzenie nowego wirtualnego środowiska języka Python z użyciem menadżera pakietów *conda* lub narzędzia *venv*.

Poniżej przedstawiono przykład stworzenia nowego środowiska conda wraz z instalacją wymaganych bibliotek z pliku *conda_requirements.txt*

```
BAM_PRO_JACH_JAWORSKI$ conda create --name <nazwa> --file conda_requirements.txt
```

Po stworzeniu środowiska należy przejść do folderu *backend* z poziomu głównego folderu repozytorium (ewentualnie aktywować stworzone wirtualne środowisko). Następnie należy skonfigurować zmienne środowiskowe w pliku *.env* w podfolderze *backend* według schematu z pliku *.env.schema*. Zmienna *HOST* powinna być zgodna z nazwą DNS serwera albo jego adresem IP. Z kolei zmienna *SECRET_KEY* jest tajnym kluczem kryptograficznym używanym przez django do generowania tokenów i szyfrowania danych. Do wygenerowania klucza należy użyć funkcji *get_random_secret_key* z *django.core.management.utils*.

Kolejnym krokiem jest dokonanie migracji bazy danych w celu utworzenia wymaganych tabel:

```
../bam_pro_jach_jaworski/backend$ python manage.py makemigrations
Migrations for 'mainApp':
  mainApp\migrations\0002_creditstorage.py
    - Create model CreditStorage

../bam_pro_jach_jaworski/backend$ python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, django_otp, mainApp, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying django_otp.0001_initial... OK
```

```
Applying mainApp.0001_initial... OK
Applying mainApp.0002_creditstorage... OK
Applying sessions.0001_initial... OK
```

Następnie można utworzyć użytkownika typu superuser, który będzie miał dostęp do strony administracyjnej django.

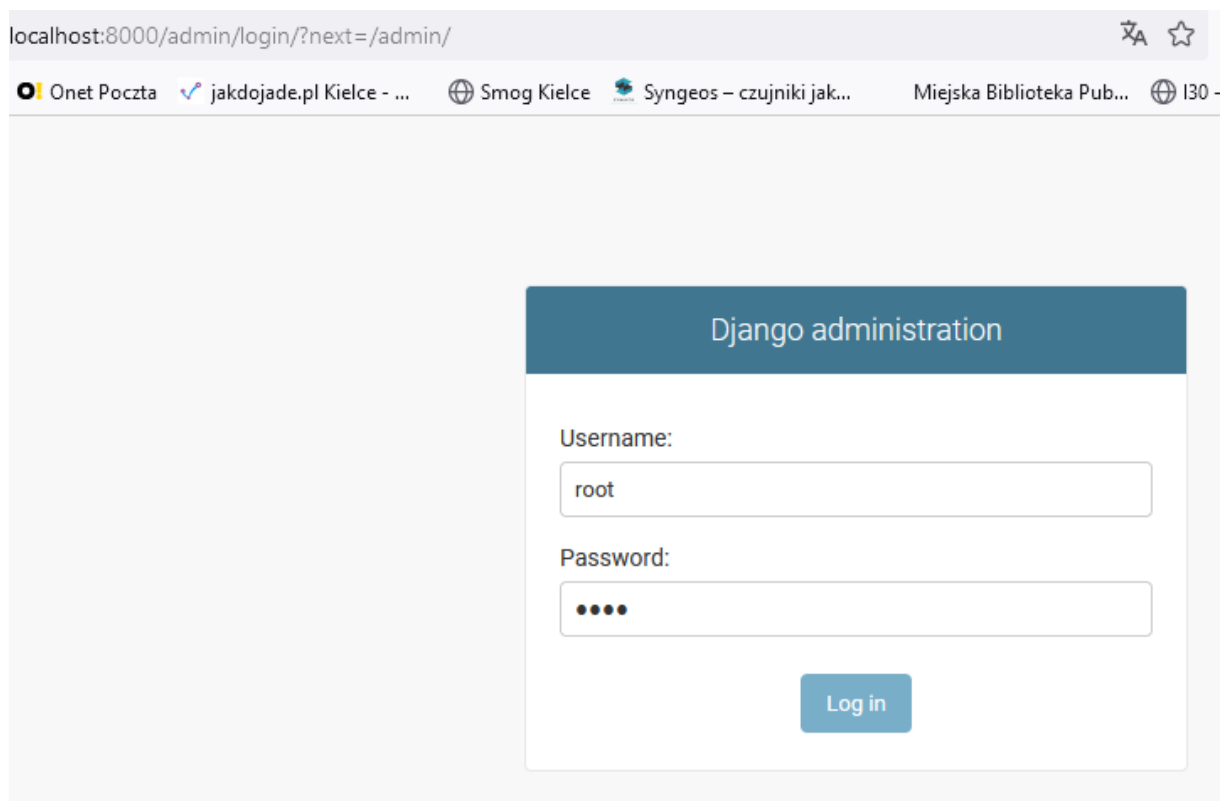
```
../bam_pro_jach_jaworski/backend$ python manage.py createsuperuser
Username (leave blank to use 'domowy'): root
Email address: root@localhost.pl
Password:
Password (again):
Superuser created successfully.
```

Po utworzeniu nowego użytkownika można uruchomić serwer backend:

```
../bam_pro_jach_jaworski/backend$ python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

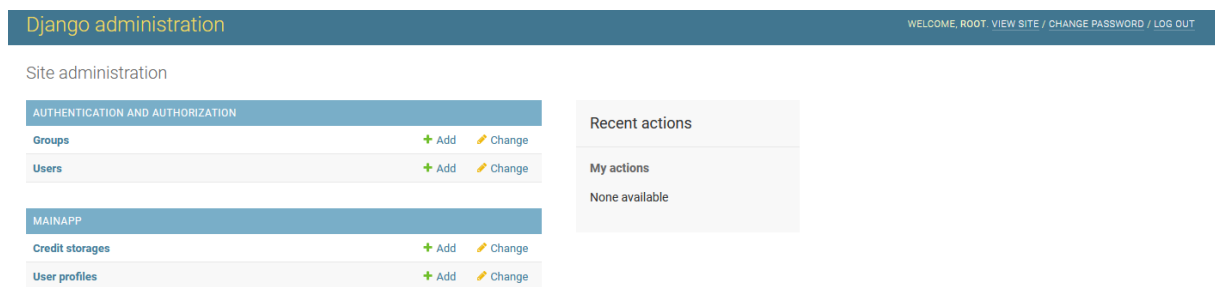
System check identified no issues (0 silenced).
December 08, 2023 - 13:22:36
Django version 4.1, using settings 'backend.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Aby uzyskać dostęp do strony administracyjnej konieczne jest otwarcie strony `/admin` i podanie danych do logowania:

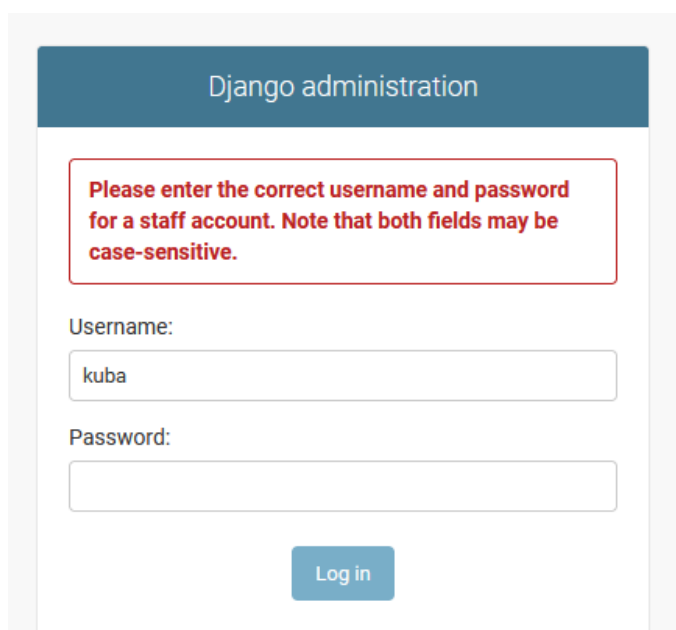


The screenshot shows a web browser window with the address bar displaying `localhost:8000/admin/login/?next=/admin/`. The browser's tab bar includes several open tabs: "Onet Poczta", "jakdojade.pl Kielce - ...", "Smog Kielce", "Syngeos – czujniki jak...", "Miejska Biblioteka Pub...", and "I30 -". The main content area of the browser shows the Django administration login interface. It features a dark blue header with the text "Django administration". Below the header, there are two input fields: "Username:" with the value "root" entered, and "Password:" with four dots indicating a masked password. A blue "Log in" button is positioned below the password field.

Po zalogowaniu pokazany zostanie główny widok strony administracyjnej



Możliwe jest zarządzanie użytkownikami, podgląd zapisanych danych i ich modyfikacja, dodawanie i usuwanie. Dostęp do strony jest możliwy jedynie dla użytkowników z uprawnieniami administratora. W przeciwnym wypadku strona odmówi dostępu.



Uruchomienie serwera z dostępem z poziomu aplikacji mobilnej

Domyślnie serwer django uruchamia się na adresie 127.0.0.1 (localhost) co uniemożliwia dostęp z zewnątrz. Aby uruchomić projekt na innym porcie należy podać parę *adres:port* podczas uruchamiania.

```
../bam_pro_jach_jaworski/backend$ python manage.py runserver 0.0.0.0:8000
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
December 08, 2023 - 13:32:46
Django version 4.1, using settings 'backend.settings'
Starting development server at http://0.0.0.0:8000/
Quit the server with CTRL-BREAK.
```

Tak uruchomiony serwer umożliwia połączenie się z nim przez aplikację mobilną. Możliwe jest również wdrożenie aplikacji za pomocą serwera WSGI jak np. [gunicorn](https://docs.djangoproject.com/en/4.1/howto/deployment/wsgi/gunicorn/).

Uruchomienie aplikacji mobilnej

Aplikację mobilną można uruchomić za pośrednictwem Expo, które udostępnia serwer deweloperski języka JavaScript (nodejs), do którego łączymy się z użyciem aplikacji Expo zainstalowanej na telefonie.

Przed uruchomieniem aplikacji konieczne jest zainstalowanie wymaganych bibliotek

```
../bam_pro_jach_jaworski/frontend$ npm i
added 1255 packages, and audited 1256 packages in 1m

73 packages are looking for funding
  run `npm fund` for details

6 moderate severity vulnerabilities

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

Po zainstalowaniu wymaganych bibliotek należy skonfigurować zmienną środowiskową *EXPO_PUBLIC_API_URL* w pliku *.env* (plik należy utworzyć ręcznie). Zmiennej tej należy przypisać wartość będącą nazwą komputera, na którym uruchomiony został serwer django lub jego adres IP i numer portu np. *EXPO_PUBLIC_API_URL=192.168.1.15:8000*

Przed kolejnymi etapami konieczne jest zalogowanie się na konto Expo:

```
../bam_pro_jach_jaworski/frontend$ npx expo login
Log in to EAS
✓ Email or username ... root@localhost.com
✓ Password ... *****
```

Po skonfigurowaniu środowiska można uruchomić serwer deweloperski aplikacji:

```
../bam_pro_jach_jaworski/frontend$ npx expo start
Starting project at (...)bam_pro_jach_jaworski\frontend
env: load .env
env: export EXPO_PUBLIC_API_URL
Starting Metro Bundler
Some dependencies are incompatible with the installed expo version:
  expo-local-authentication@13.6.0 - expected version: ~13.4.1
  react-native@0.72.5 - expected version: 0.72.6
Your project may not work correctly until you install the correct versions of the
packages.
Fix with: npx expo install --fix
```



A square QR code with a black and white pixelated pattern, used for scanning with the Expo Go app on a mobile device.

```
> Metro waiting on exp://192.168.1.15:8081
> Scan the QR code above with Expo Go (Android) or the Camera app (iOS)

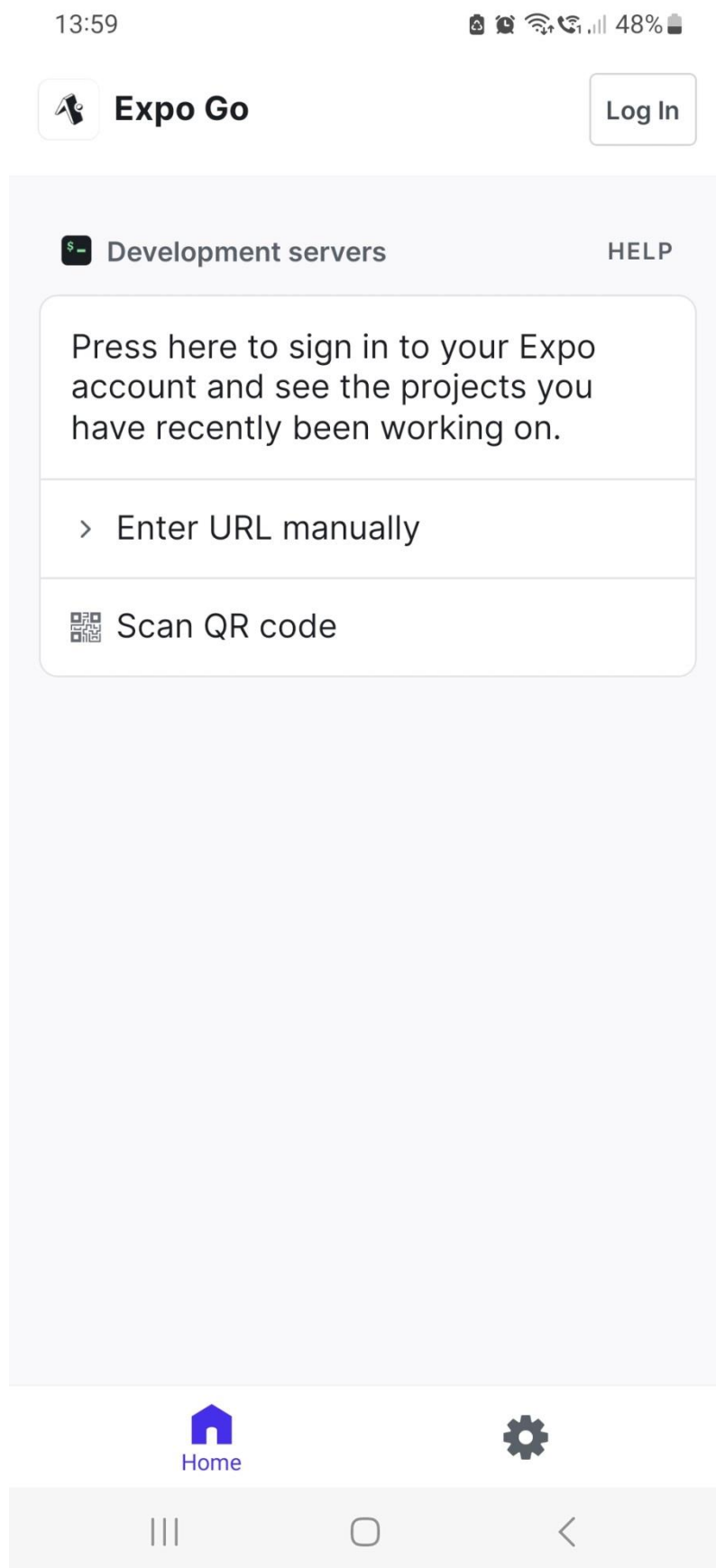
> Using Expo Go
> Press s | switch to development build

> Press a | open Android
> Press w | open web

> Press j | open debugger
> Press r | reload app
> Press m | toggle menu
> Press o | open project code in your editor

> Press ? | show all commands
```

Aplikacja Expo Go otwarta na rzeczywistym urządzeniu.



Po zeskanowaniu kodu QR rozpocznie się „bundlowanie” aplikacji:

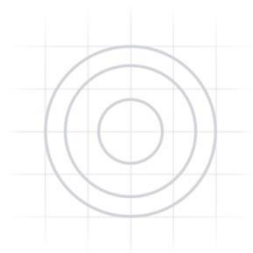
Android node_modules\expo\AppEntry.js 8.6% (120/486)

Z kolei na ekranie telefonu również ukaże się stosowna informacja:

13:59

48%

Bundling 2,7%...

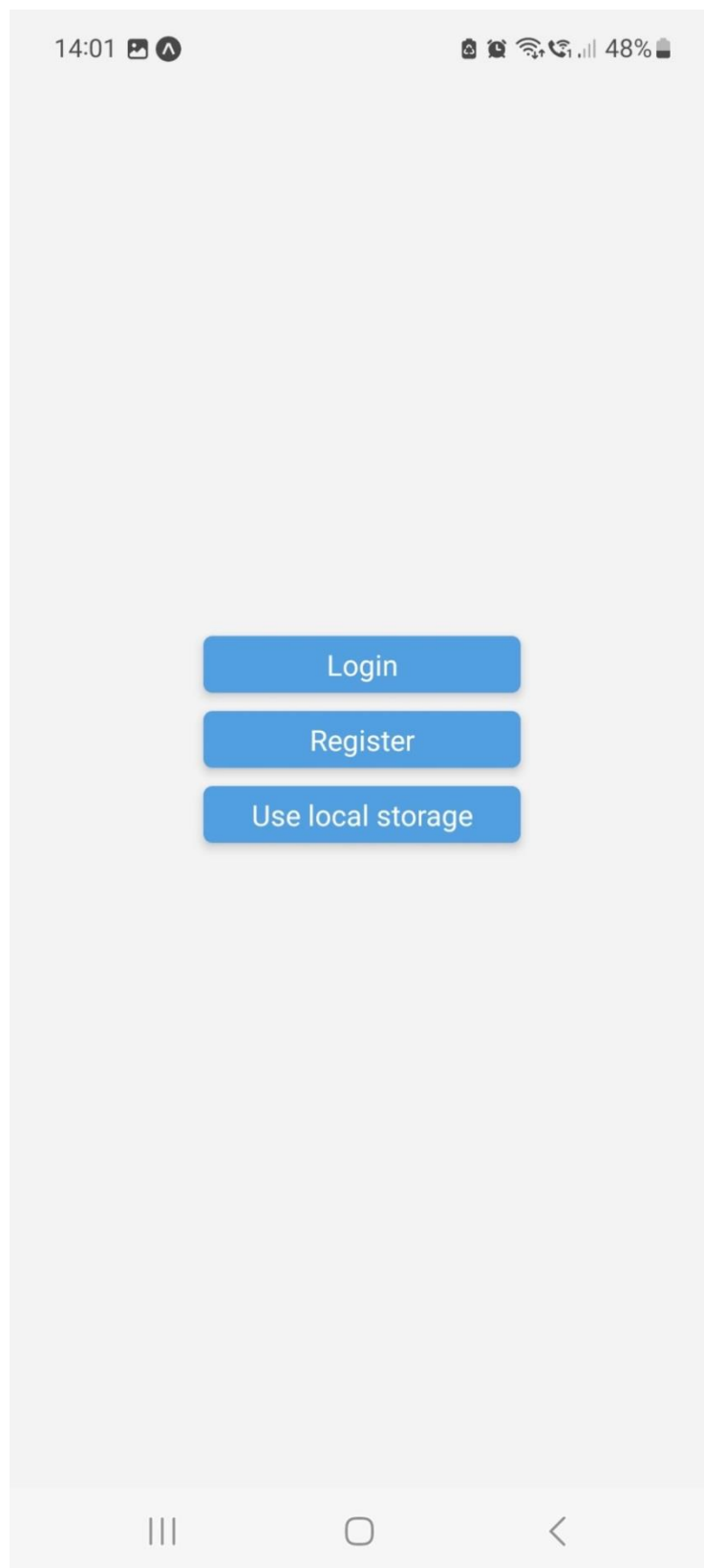


Bundling

2,74%



Po zbudowaniu aplikacji zostanie ona otwarta na telefonie:



Z kolei w konsoli pojawi się komunikat o zakończeniu budowania aplikacji:

```
Android Bundling complete 39161ms
```

Budowanie aplikacji do dystrybucji

Po przetestowaniu aplikacji możliwe jest zbudowanie jej do pliku *.aab* lub *.apk* w celu zainstalowania na systemie Android (na przykładzie lokalnej kompilacji do pliku *.apk*):

```
../frontend$: eas build --platform android --local --profile preview
Loaded "env" configuration for the "preview" profile: no environment variables
specified. Learn more: https://docs.expo.dev/build-reference/variables/
✓ Using remote Android credentials (Expo server)
✓ Using Keystore from configuration: Build Credentials 2cYxBCzwAP (default)
✓ Compressed project files 12s (31.3 MB)
(...pominięto...)
[PREBUILD] - Config syncing
[PREBUILD] ✓ Config synced
[PREBUILD] Running "npm install" in /tmp/jakub/eas-build-local-nodejs/60fae58b-
2888-4492-bb9e-8c6721f66896/build/frontend directory
[PREBUILD] added 1 package, and audited 1258 packages in 3s
[PREBUILD]
[PREBUILD] 74 packages are looking for funding
[PREBUILD] run `npm fund` for details
[PREBUILD] 7 moderate severity vulnerabilities
[PREBUILD]
[PREBUILD] To address issues that do not require attention, run:
[PREBUILD]   npm audit fix
[PREBUILD]
[PREBUILD] To address all issues (including breaking changes), run:
[PREBUILD]   npm audit fix --force
[PREBUILD]
[PREBUILD] Run `npm audit` for details.
[PREPARE_CREDENTIALS] Writing secrets to the project's directory
[PREPARE_CREDENTIALS] Injecting signing config into build.gradle
[RUN_GRADLEW] Running 'gradlew :app:assembleRelease' in /tmp/jakub/eas-build-
local-nodejs/60fae58b-2888-4492-bb9e-8c6721f66896/build/frontend/android
[RUN_GRADLEW] Starting a Gradle Daemon, 1 incompatible and 1 stopped Daemons could
not be reused, use --status for details
(...pominięto...)
[RUN_GRADLEW] Using expo modules
[RUN_GRADLEW]   - expo-application (5.3.1)
[RUN_GRADLEW]   - expo-clipboard (4.3.1)
[RUN_GRADLEW]   - expo-constants (14.4.2)
[RUN_GRADLEW]   - expo-file-system (15.4.4)
[RUN_GRADLEW]   - expo-font (11.4.0)
[RUN_GRADLEW] - expo-keep-awake (12.3.0)
[RUN_GRADLEW]   - expo-local-authentication (13.6.0)
[RUN_GRADLEW]   - expo-modules-core (1.5.11)
[RUN_GRADLEW]   - expo-modules-core$android-annotation (1.5.11)
[RUN_GRADLEW]   - expo-modules-core$android-annotation-processor (1.5.11)
[RUN_GRADLEW]   - expo-secure-store (12.3.1)
[RUN_GRADLEW]   - expo-splash-screen (0.20.5)
[RUN_GRADLEW] > Configure project :react-native-reanimated
```

```
[PREPARE_ARTIFACTS] Writing artifacts to
(...)/bam_pro_jach_jaworski/frontend/build-1702042231017.apk
```


Build successful

You can find the build artifacts in
(...)/bam_pro_jach_jaworski/frontend/build-1702042231017.apk

Tak utworzoną aplikację można następnie udostępnić do zainstalowania lub przetestować ją na emulatorze instalując ją za pomocą adb np.:


```
../frontend$: adb install build-1702042231017.apk
Performing Streamed Install
Success
```

Możliwe jest również zbudowanie aplikacji korzystając z serwerów Expo:

 Android Play Store build 1.0.0 (1) less than a minute ago	69f0d91*	production	None	None
--	----------	------------	------	------



Wybierając build możliwe jest śledzenie postępu:


Builds > ad728d7f


 **Android Play Store build**


69f0d91 · Docs for frontend


Show Details


Profile	SDK version	Version	Version code	Commit	Created by
production	49.0.0	1.0.0	1	69f0d91* 	 kubaj


 **Build artifact**


Notify me 







 **Status: Build in progress**

Elapsed time:  58s


 **Logs**




>  Waiting to start  9s


<  Spin up build environment  48s


1 Creating new worker instance

 **Build artifact**

AAB

Download 

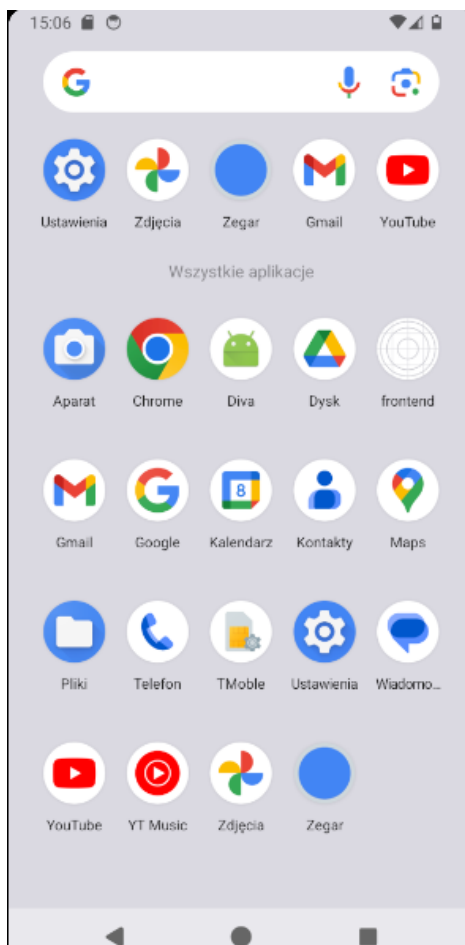
Status	Start time	Wait time	Queue time	Build time	Total time	Availability
 Finished	Dec 8, 2023 2:23 PM	None	55s	8m 49s	9m 45s	29 days

Status	Start time	Wait time	Queue time	Build time	Total time	Availability
 Finished	Dec 8, 2023 2:23 PM	None	55s	8m 49s	9m 45s	29 days

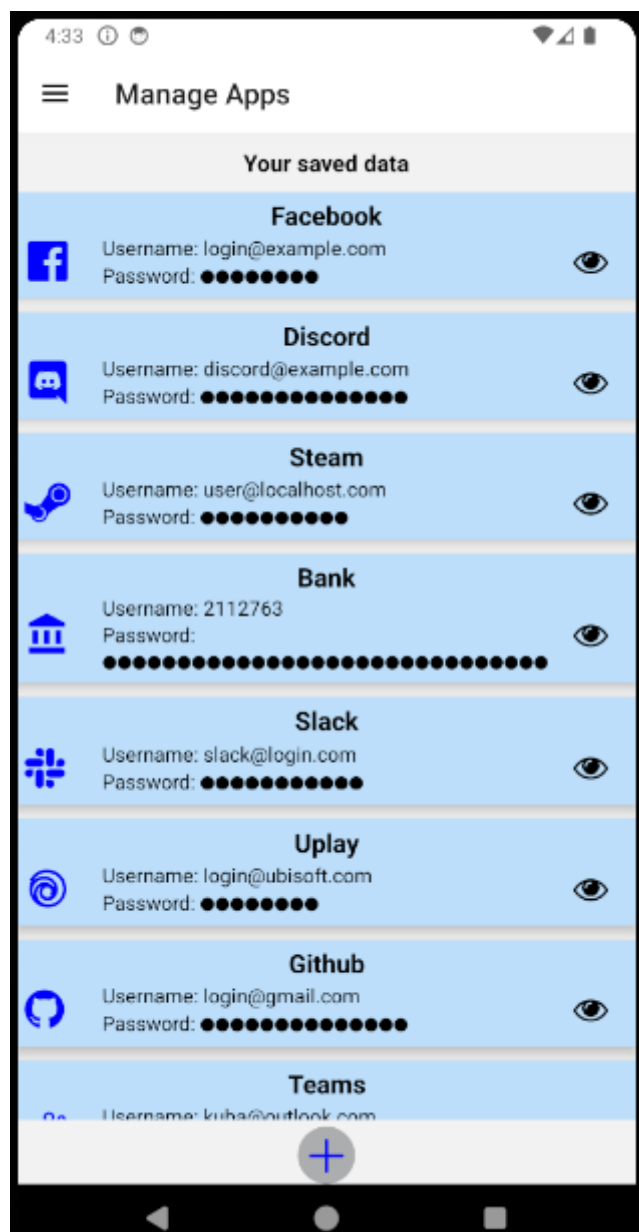
Po zbudowaniu aplikacji możliwe jest pobranie jej z serwera i zainstalowanie na emulatorze albo telefonie

```
(...)  
Waiting in Free tier queue  
|███████████████████████████████████████████████████████████████████████████|  
  
✓ Build finished  
📄 Android app:  
https://expo.dev/artifacts/eas/b7Yp1LZneyWutfyoBDsnKD.apk  
  
✓ Install and run the Android build on an emulator? ... yes  
  
✓ Successfully downloaded app 1s  
  
Using open emulator: Pixel_3a  
  
Installing your app...  
✓ Successfully installed your app!  
  
Starting your app...  
✓ Successfully started your app!
```

Aplikacja została zainstalowana na emulatorze



Główne okno aplikacji z przykładowymi danymi aplikacji uruchomionej na emulatorze.



Informacje na temat udostępniania aplikacji w sklepie Play dostępne są na podanych stronach:

- <https://developer.android.com/studio/publish>
- <https://docs.expo.dev/submit/introduction/>

Dokumentacja części backend aplikacji dostępna jest [tutaj](#).

Dokumentacja aplikacji mobilnej dostępna jest [tutaj](#).