

Praca konkursowa

Odkryj Rudy

Autor: Bartłomiej Pacia, barpac02@gmail.com

Opiekun: Donat Dubiel, knei.pracownia@gmail.com

Koło Naukowe Elektroniki i Informatyki w Knurowie

ul. Batorego 4, 44-194 Knurów

SPIS TREŚCI

1	Wprowadzenie	1
2	Przeznaczenie projektu	1
3	Opis rozwiązania	2
3.1	Zarys techniczny	2
3.2	Aplikacja mobilna.....	2
3.3	Funkcje serwerowe.....	3
3.4	Turystyczna Baza Danych.....	3
4	Wyniki	5
4.1	Współpraca z nadleśnictwem	5
4.2	Współpraca z gminą.....	5
4.3	Odbiór społeczny	5
5	Plany na przyszłość.....	5
5.1	Powiększenie liczby regionów	5
5.2	Program lojalnościowy	5
5.3	Inne funkcje	6
6	Źródła	6

1 WPROWADZENIE

Rudy to licząca około trzy tysiące mieszkańców miejscowość położona w powiecie raciborskim, w gminie Kuźnia Raciborska. Pierwsze wzmianki o Rudach datuje się na rok 1258, kiedy to nad brzegiem rzeki Cystersi wybudowali swoje opactwo. Obecnie Rudy są ważnym punktem na turystycznej mapie województwa śląskiego, znanym ze swoich walorów historycznych i przyrodniczych. Otoczone są Lasami Rudzkimi – jednym z największych kompleksów leśnych na Górnym Śląsku, bogatym w liczne gatunki drzew i zwierząt, pełnym uroczysk i innych ciekawych miejsc, do których dojechać można wytyczonymi ścieżkami rowerowymi.

2 PRZEZNACZENIE PROJEKTU

Odkryj Rudy to system informacji turystycznej o miejscowości Rudy i okolicach. W jego skład wchodzi aplikacja mobilna na iOS i Androida, serwer w chmurze, baza danych oraz zestaw narzędzi usprawniających poszczególne etapy pracy z systemem. W związku z rosnącym zainteresowaniem w roku 2020 został rozszerzony na obszar całej Gminy Kuźnia Raciborska.

Obecnie w aplikacji umieszczono opisy ponad 70 miejsc wraz ze zdjęciami, historią, ciekawostkami, współzrędnymi geograficznymi i innymi właściwościami. Zostały one podzielone na kategorie: zabytki, ciekawe miejsca, atrakcje, kamienie pamiątkowe i świetlice wiejskie. Użytkownicy mogą dodawać do miejsc polubienia (popularne „lajki”) oraz komentarze ze zdjęciami, a jeśli są w pobliżu miejsca (bliżej niż 100 m), mają możliwość zaznaczenia miejsca jako odkryte. Dodatkowo administrator systemu może wysyłać na urządzenia z zainstalowaną aplikacją powiadomienia, aby np. poinformować o nadchodzącym wydarzeniu kulturalnym.



Strona internetowa projektu: <https://odkryjrudy.pl>

Ryc. 1. Obecna ikona aplikacji.

3 OPIS ROZWIĄZANIA

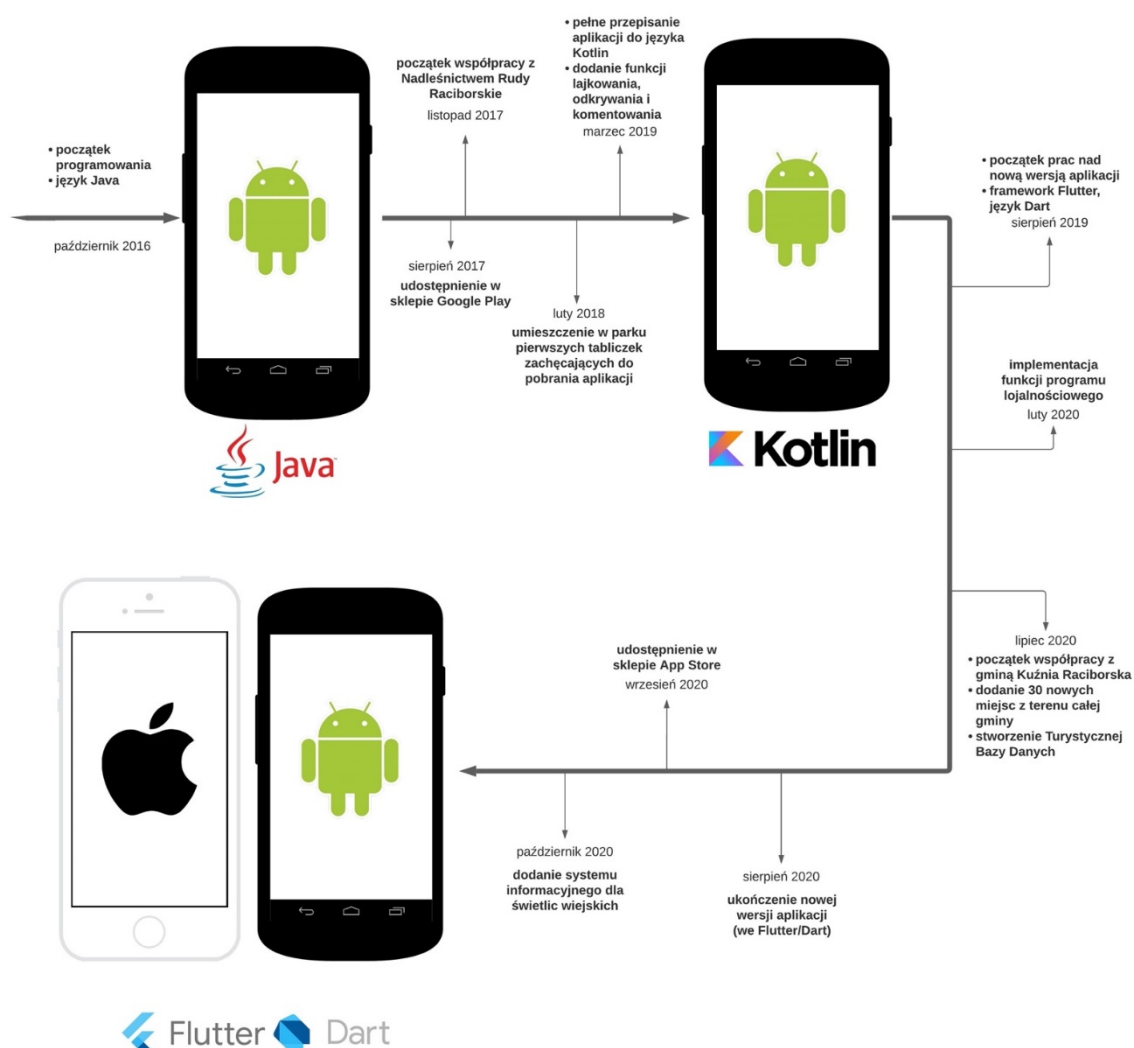
3.1 ZARYS TECHNICZNY

Odkryj Rudy składa się z 3 głównych komponentów:

- aplikacji mobilnej
- funkcji serwerowych
- Turystycznej Bazy Danych

3.2 APLIKACJA MOBILNA

Aplikacja mobilna była pierwszym (i przez długi czas jedynym) komponentem systemu, a jednocześnie tym, który przeszedł najwięcej zmian (ryc. 2). Obecnie napisana jest w całości w języku programowania Dart we frameworku Flutter. Największą zaletą takiego rozwiązania jest możliwość stworzenia wydajnego, natywnego programu na iOS i Androida z jednego kodu źródłowego.



Ryc. 2. Wybrane wydarzenia z procesu rozwijania aplikacji.

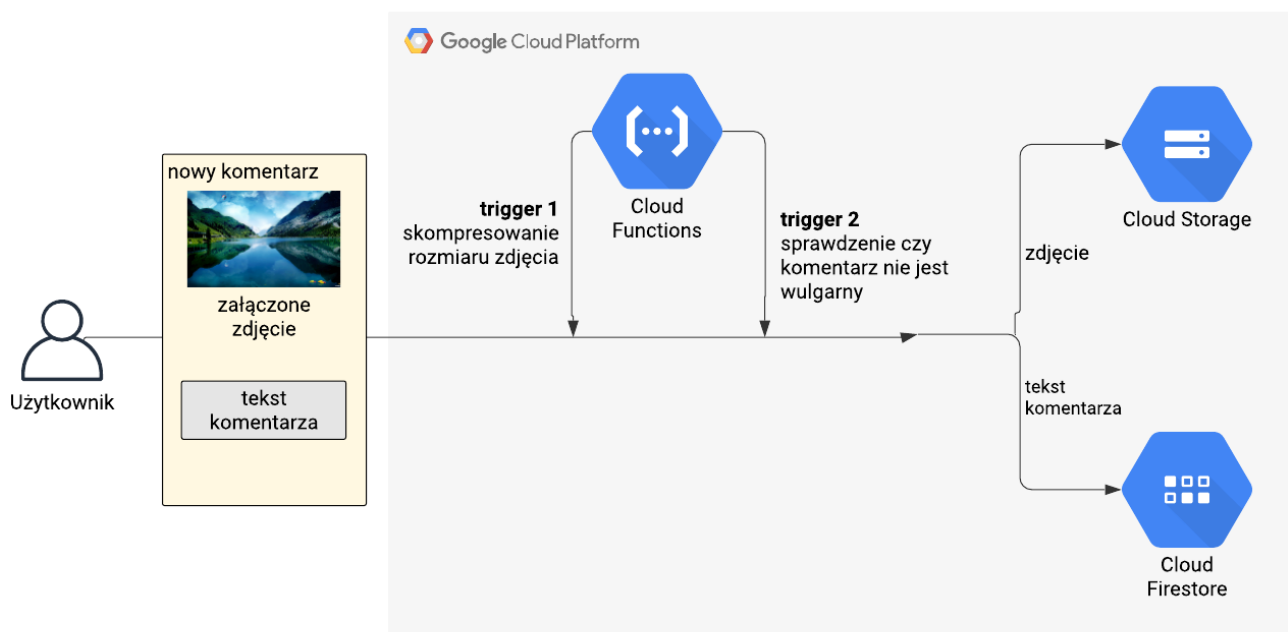
3.3 FUNKCJE SERWEROWE

Odkryj Rudy nie posiada własnego serwera w rozumieniu tradycyjnym, ale zamiast niego, do uruchamiania kodu serwerowego korzysta z usługi Google Cloud Functions. Pozwala ona na uruchamianie kodu serwerowego w chmurze, tj. na serwerach automatycznie zarządzanych przez Google. Pozwala to zminimalizować koszty i czas wymagane na zarządzanie własnym serwerem. Upraszcza też wiele kwestii związanych z bezpieczeństwem i czynnościami utrzymaniowymi (aktualizowanie oprogramowania, optymalizowanie wydajności).

Funkcje były początkowo napisane w języku JavaScript, ale z powodu jego wad (brak statycznego typowania) potęgowanych przez szybko rosnącą ilość kodu zostały przepisane do TypeScript.

Funkcje serwerowe w Odkryj Rudy można podzielić na 2 rodzaje:

- **triggery** – ich zadaniem jest uruchamianie kodu w odpowiedzi na jakąś akcję podjętą przez użytkownika. Na przykład, gdy użytkownik doda do miejsca komentarz ze zdjęciem, funkcje sprawdzają, czy komentarz nie jest wulgarny i kompresują zdjęcie (ryc. 3).



Ryc. 3. Wybrane wydarzenia z procesu rozwijania aplikacji

- **funkcje zaplanowane** – ich zadaniem jest przeprowadzanie czynności utrzymaniowych – np. operacji, które muszą być uruchamiane w konkretnych interwałach (np. aktualizacja co 15 minut danych dotyczących jakości powietrza, pozbywanie się zdjęć będących częścią usuniętych komentarzy).

Użytkownicy Odkryj Rudy mogą tworzyć kilka rodzajów obiektów, między innymi polubienia miejsc, wizyty oraz komentarze (wraz z załączonymi do nich zdjęciami). Wszystkie te dane przechowywane są w Cloud Firestore – nierelacyjnej bazie danych (noSQL), zapewnianej przez platformę Google Firebase oraz Cloud Storage. Główną zaletą Cloud Firestore jest możliwość odpytywania bazy w czasie rzeczywistym. Dane umieszcza się naprzemiennie w kolekcjach i dokumentach, co swoją strukturą przypomina drzewo systemu plików.

3.4 TURYSTYCZNA BAZA DANYCH

Jest to najmłodsza część systemu, stworzona w lecie 2020 roku. Jej zadaniem jest ustandaryzowanie formatu przechowywania danych oraz ogólne usprawnienia procesu pracy z nimi. Kluczowym pojęciem w Bazie jest paczka. Paczka to po prostu folder o określonej strukturze zawierający zdjęcia, teksty i inne dane (np. pliki JSON) opisujące dany rejon (np. gminę).

Paczka to pojęcie trochę abstrakcyjne, ale dobrze oddające jej istotę. Tak naprawdę istnieją 2 rodzaje paczek:

- **Paczka źródłowa** to ściśle określona struktura folderów i plików (drzewo), zawierająca wszystkie zebrane dane o danym rejonie (zdjęcia w wysokiej jakości, teksty w kilku językach). Strukturę paczki źródłowej dla Rud można zobaczyć w pliku *struktura.txt*.

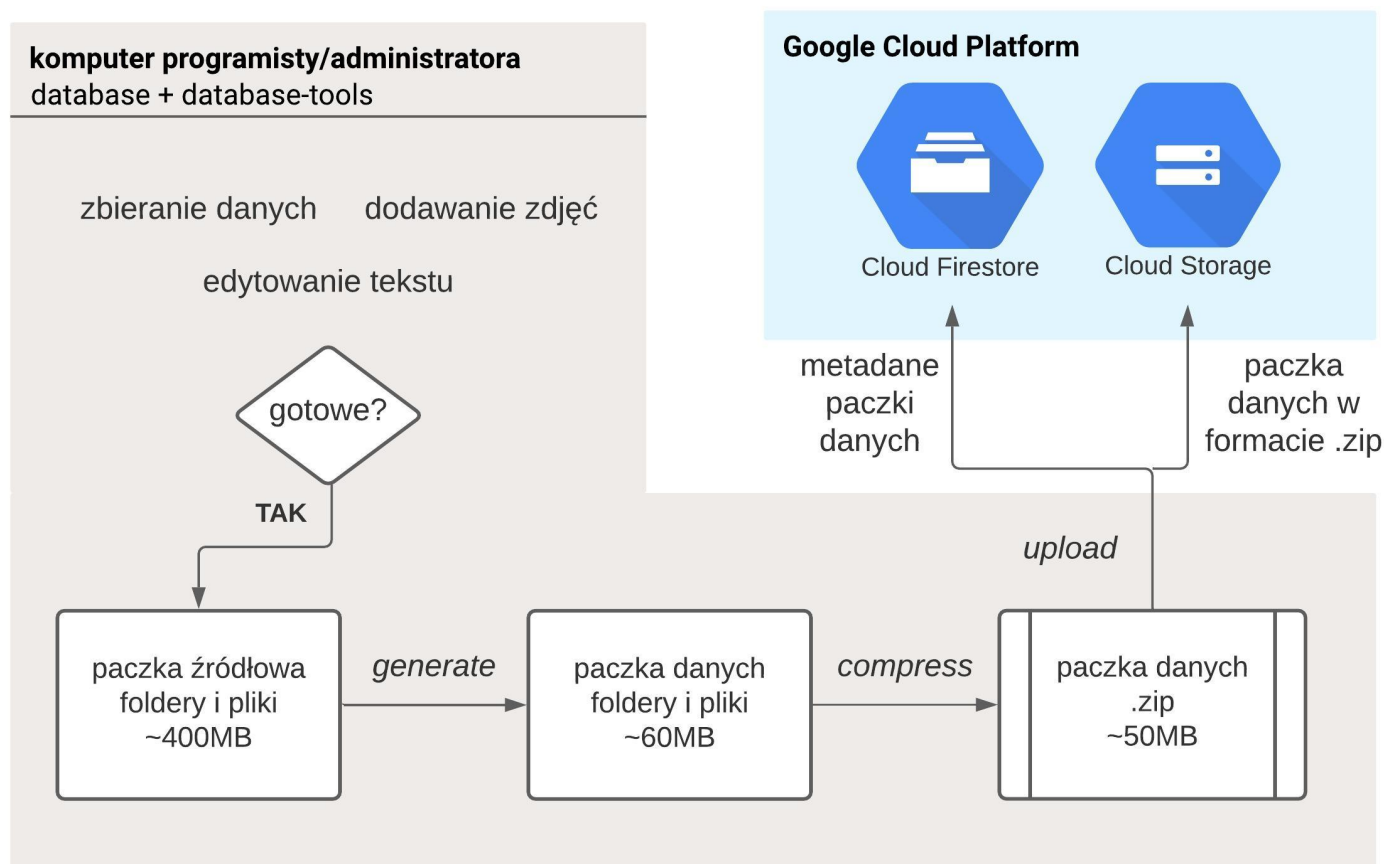
- *Paczka danych* to ogólne określenie na folder lub plik .zip zawierający tekstowe i graficzne dane, gotowe do użycia przez aplikację.

Dlaczego to rozróżnienie pomiędzy *paczkami źródłowymi* a *paczkami danych*?

Paczka źródłowa może zawierać np. 10 zdjęć jakiegoś miejsca w wysokiej jakości. Jednak bezpośrednie użycie tych 10 zdjęć w aplikacji mobilnej oznaczałoby marnotrawienie zasobów użytkowników (transfer internetu, pamięć), którzy byliby zmuszeni do ich pobierania na swoje urządzenia. Dlatego tworzy się *paczkę danych*. Paczki danych to mniejsze, skompresowane wersje *paczek źródłowych*, które zawierają np. tylko 4 ze wspomnianych 10 zdjęć i to mocno skompresowanych. Ich jakość będzie zupełnie wystarczająca do wyświetlenia na smartfonie, a jednocześnie będą zajmować od kilku do kilkunastu razy mniej pamięci.

Z *paczki źródłowej* można stworzyć *paczkę danych* za pomocą programów zawartych w projekcie *database-tools*.

Zawiera on kilka prostych programów upraszczających cały proces pracy z bazą danych (ryc. 4)



Ryc. 4. Wybrane wydarzenia z procesu rozwijania aplikacji

- program *generate* generuje paczkę danych. Przyjmuje kilka argumentów, z których najważniejsze to ID paczki oraz język, w którym będą treści tekstowe paczki danych (oczywiście o ile takowe przetłumaczone teksty są dostępne w paczce źródłowej). Po uruchomieniu program chodzi po drzewie folderów i plików paczki źródłowej (jego struktura znajduje się w pliku `drzewo_paczki_1.txt`), wybiera odpowiednie zdjęcia i pliki tekstowe i tworzy z nich folder paczki danych.
- program *compress* kompresuje paczkę danych z postaci folderu wyjściowego do pliku .zip
- program *upload* przesyła paczkę danych w formacie .zip do Cloud Storage oraz odpowiednie metadane do Cloud Firestore. Natychmiast po przesłaniu stają się one widoczne dla użytkowników aplikacji.

W grudniu 2020 roku *database-tools* zostały przepisane z Pythona do języka Go, dzięki czemu wzrosła niezawodność i szybkość działania programu. Dodatkowo struktura kodu została gruntownie zrefaktoryzowana, dzięki czemu dodawanie nowych funkcji do programów w przyszłości będzie prostsze.

4 WYNIKI

Można powiedzieć, że projekt odniósł sukces. Do połowy stycznia 2020 roku aplikację pobrało prawie dwa tysiące użytkowników (ok. 1950. Źródło: usługa Firebase Analytics i sklep Google Play).

4.1 WSPÓŁPRACA Z NADLEŚNICTWEM

W 2017 roku nawiązana została współpraca z Nadleśnictwem Rudy Raciborskie. Nadleśnictwo zamieściło na swojej stronie internetowej informację o aplikacji oraz zakupiło i przekazało 50 aluminiowych tabliczek z nadrukowanym kodem QR, zachęcających do pobrania aplikacji. Obecnie większość z nich jest zamieszczona przy ciekawych miejscach w przyklasztornym parku oraz na całym terenie Lasów Rudzkich (ryc. 5).

Ryc. 5. Tablica informacyjna z przykręconym kodem QR i zachętą do pobrania aplikacji.



4.2 WSPÓŁPRACA Z GMINĄ

W 2020 roku nawiązana została współpraca z Gminą Kuźnia Raciborska (zob. załącznik gmina.jpg). Obszar zawarty w aplikacji został rozszerzony na teren całej gminy. Aplikacja otrzymała również wsparcie finansowe, dzięki czemu możliwy był zakup licencji w Apple Developer Program (\$99 rocznie) i opublikowanie aplikacji w sklepie App Store. Do systemu dodany został również prosty system informacyjny, który ułatwia komunikację pomiędzy instruktorami świetlic wiejskich (czyli osobami, które je prowadzą) i osobami, które korzystają z oferty świetlic wiejskich. Nie był on jednak jeszcze używany w praktyce, gdyż został ukończony akurat, gdy zaczęła się druga fala pandemii i wszystkie wiejskie świetlice zostały zamknięte.

4.3 ODBIÓR SPOŁECZNY

Aplikacja jest pozytywnie odbierana przez użytkowników, o czym świadczy średnia ocen 4,9/5 oraz pozytywne recenzje w sklepie Google Play. W załącznikach (zob. pliki kom_*.jpg) znajduje się kilka wybranych recenzji, a plik mail.png zawiera ciekawy e-mail, jaki otrzymałem od najaktywniejszego użytkownika aplikacji po poproszeniu go o podzielenie się opinią i wrażeniami.

5 PLANY NA PRZYSZŁOŚĆ

Odkryj Rudy jest dojrzałym projektem, zbudowanym na stabilnym stosie technologicznym, a architektura całego systemu dobrze sprawdza się w praktyce. Stwarza to szerokie pole do rozwoju aplikacji.

5.1 POWIĘKSZENIE LICZBY REGIONÓW

Oczywistym polem do rozwoju dla aplikacji typu Odkryj Rudy jest poszerzenie obszaru, który w sobie zawiera. Struktura paczki danych jest prosta, a zarazem efektywna – wystarczy zabrać dane (zdjęcia ciekawych miejsc i zabytków danego regionu, ich lokalizacje i historię), a następnie umieścić je w odpowiedniej strukturze folderów i plików – dzięki czemu program generat z database-tools będzie w stanie stworzyć paczkę danych gotową do użycia przez aplikację. Cechą wyróżniającą Odkryj Rudy jest otwarty format paczki. Dzięki temu będzie mieć szansę na stanie się tanim i dobrym wyborem dla gmin/rejonów, które podobnie jak Rudy 4 lata temu, nie mają zdigitalizowanej bazy danych swojej oferty turystycznej.

5.2 PROGRAM LOJALNOŚCIOWY

Ciekawym pomysłem jest program lojalnościowy. W ramach takiego programu użytkownicy mogliby odbierać określone nagrody za odkryte miejsca. Na przykład, jeśli użytkownik odkrył 10 miejsc, to w Punkcie Informacji Turystycznej w Rudach może za darmo odebrać album ze zdjęciami, a jeśli jest naprawdę zapalonym turystą i odkrył np. 50 miejsc, to może odebrać T-shirt z napisem „Odkryj Rudy” i bidon na rower.

Prace nad programem lojalnościowym od strony technicznej zostały zakończone na początku 2020 roku. W toku były rozmowy z Nadleśnictwem Rudy Raciborskie dot. szczegółów nagród i kwestii prawnych. Niestety, z powodu pandemii i innych wydarzeń realizacja programu lojalnościowego we współpracy z Nadleśnictwem stała się niemożliwa. Dlatego

też zwróciłem się z tą kwestią do gminy Kuźnia Raciborska i otrzymałem pozytywną odpowiedź. Prace nad programem trwają.

5.3 INNE FUNKCJE

5.3.1 automatyczne odkrywanie miejsc

Po włączeniu odpowiedniej opcji, aplikacja będzie automatycznie miejsca jako odkryte, gdy użytkownik jest w pobliżu.

5.3.2 stworzenie wersji internetowej

Użytkownicy mieliby dostęp do większości funkcji aplikacji mobilnej z poziomu przeglądarki internetowej. Framework Flutter w przyszłości pozwoli na budowanie projektu nie tylko na iOS i Androida, ale też do wersji webowej.

5.3.3 dodanie testów jednostkowych i integracyjnych

Obecnie to ja jestem jedną osobą pracującą nad systemem i znam go „na wylot”, więc brak testów nie jest dużym problemem. Jednak w przypadku, gdy w przyszłości nad Odkryj Rudy pracowałoby parę osób, testy znacząco uproszczą proces rozwijania aplikacji i dadzą programistom pewność, że kod, który piszą, działa.

5.3.4 kompletna przebudowa części serwerowej

Zrezygnowanie z funkcji serwerowych na rzecz „normalnego” REST API. Stworzone byłoby ono w sposób modularny, z uwzględnieniem wykorzystania nowoczesnych metod konteneryzacji (Docker, Kubernetes). W ten sposób w przyszłości będzie można efektywnie skalować liczbę instancji poszczególnych komponentów w zależności od obciążenia (na wypadek nagłego napływu użytkowników).

5.3.5 Udostępnienie bazy danych

Jako wielki entuzjasta oprogramowania open-source uważam, że każdy mieszkaniec powinien mieć wolny i darmowy dostęp do wszystkich danych o swojej okolicy, w możliwie najprostszy sposób. Dlatego ważną dla mnie kwestią jest publiczne udostępnienie Turystycznej Bazy Danych (projekty database i database-tools). Na razie nie jest jeszcze to możliwe z powodu szczątkowej dokumentacji oraz kwestii prawnych, które wymagają przeanalizowania, ale jestem zdeterminowany, aby to uczynić.

5.3.6 Nowa nazwa

Pomniejszą, ale ważną z marketingowego punktu widzenia jest nazwa aplikacji i całego systemu. Obecnie pełna, oficjalna nazwa systemu to Odkryj Rudy i okolice, ale na potrzeby tego tekstu pozwoliłem sobie skracać ją do Odkryj Rudy.

Gdy proponowane wyżej ulepszenia wejdą w życie (w szczególności, gdy w aplikacji dostępna będzie większa liczba regionów), nazwa Odkryj Rudy przestanie być odpowiednia. Aby wybrać nową nazwę, potrzeba dobrze znać zakres projektu, by nie być zmuszonym do kolejnego rebrandingu w krótkim czasie. Na razie jednak Odkryj Rudy i okolice spełnia swoje zadanie.

6 ŹRÓDŁA

Jako że projekt jest nieustannie rozwijany od 4 lat, niemożliwym jest wymienienie wszystkich wykorzystanych źródeł, materiałów i bibliotek. Jednak nieocenioną pomocą były oficjalne dokumentacje języków programowania i narzędzi programistycznych, z których korzystałem.

- strona internetowa i dokumentacja systemu Android <https://developer.android.com>
- strona internetowa i dokumentacja platform firmy Apple <https://developer.apple.com>
- strona internetowa i dokumentacja frameworka Flutter <https://flutter.dev>
- strona internetowa i dokumentacja języka programowania Dart <https://dart.dev>
- strona internetowa i dokumentacja języka programowania Kotlin <https://kotlinlang.org>
- strona internetowa i dokumentacja języka programowania TypeScript <https://www.typescriptlang.org>
- strona internetowa i dokumentacja języka programowania Go <https://golang.org>
- strona internetowa i dokumentacja platformy Google Firebase <https://firebase.google.com>
- strona internetowa i dokumentacja Google Cloud Platform <https://cloud.google.com>
- forum dla programistów StackOverflow <https://stackoverflow.com>
- wszystkie użyte dane historyczne (książki, zdjęcia) są wymienione w aplikacji na ekranie „O aplikacji”