

Dokumentacja powdrożeniowa systemu Freeturilo

Mikołaj Ryll, Mikołaj Terzyk

Promotor: dr Paweł Rzążewski

13.01.2021

Spis treści

1 Historia zmian	2
2 Abstrakt	2
3 Dokumentacja wdrożenia	2
4 Instrukcja instalacji	3
5 Dokumentacja techniczna	4
6 Instrukcja obsługi	5
6.1 Mapa	6
6.2 Trasa	10
6.3 Historia	12
6.4 Panel administratora	13

1 Historia zmian

Tabela 1: Historia zmian

Data	Autor	Opis	Wersja
09.01.2022r.	Mikołaj Ryll, Mikołaj Terzyk	Pierwsza wersja dokumentacji	1.0

2 Abstrakt

Dokument stanowi dokumentację powdrożeniodoceną systemu *Freeturilo*. W sekcji 3. spisana została dokumentacja wdrożenia systemu. Rozdział 4. stanowi instrukcję instalacji systemu *Freeturilo*, a 5. jego dokumentację techniczną. W dziale 6. znaleźć można instrukcję obsługi systemowego interfejsu użytkownika - aplikacji mobilnej.

3 Dokumentacja wdrożenia

W architekturze systemu *Freeturilo* wyróżnić można cztery główne moduły:

1. bazę danych *PostgreSQL*, przechowującą w szczególności stacje rowerowe i szczegóły tras między nimi,
2. serwer webowy *Freeturilo Web API*, udostępniający systemowe API, a także odpowiadający za większość logiki systemu,
3. serwis *NextBike Service*, aktualizujący dane dotyczące stacji rowerowych na podstawie informacji pobieranych w czasie rzeczywistym z serwerów *NextBike*,
4. aplikację mobilną *Freeturilo* stanowiącą interfejs użytkownika.

Moduły 1-3 zostały wdrożone jako zasoby chmurowe w ramach usług dostawcy *Microsoft Azure*.

Baza danych Baza danych umieszczona została w chmurze w ramach *Azure Database for PostgreSQL*, czyli usługi *DBaaS (Database-as-a-Service)* (pochodnej *SaaS*), która udostępnia system *PostgreSQL* w wersji 11. Dobrane przez nasz zespół parametry usługi to 50 GB pamięci masowej oraz procesor jednordzeniowy z dostępem do 2 GB pamięci operacyjnej.

Serwer webowy Moduł *Freeturilo Web API* został wdrożony w chmurze z wykorzystaniem usługi *PaaS Azure App Service*, jako aplikacja webowa zaimplementowana z wykorzystaniem framework'u *ASP.NET*. Zasoby przydzielone wdrożonemu serwisowi to 1 GB pamięci masowej oraz procesor jednordzeniowy

z dostępem do 1 GB pamięci operacyjnej. Systemowe API zawarte w tym module jest dostępne w sieci publicznej pod adresem <https://freeturilowebapi.azurewebsites.net>.

Serwis Wdrożenie *NextBike Service* odbyło się w sposób analogiczny do wdrożenia *Freeturilo Web API*, z takimi samymi parametrami. Jedyną istotną różnicą jest niedostępność serwisu z poziomu sieci publicznej. Jest to moduł, który nie udostępnia API, ani żadnych innych funkcjonalności, które powinny być dostępne dla podmiotów zewnętrznych.

Aplikacja mobilna Aplikacja została zaimplementowana w wersji 8. języka *Java* i jest kompatybilna z systemem *Android* o wersji 6.0 (*Marshmallow*) lub wyższej. W przeciwieństwie do powyższych modułów, wdrożona została jako plik *APK* (*Android Package Kit*). Format ten używany jest do dystrybucji i instalacji pakietów systemu operacyjnego *Android*.

4 Instrukcja instalacji

Instalacja każdego z modułów systemu odbywa się niezależnie. Moduły, które zostały wdrożone jako zasoby chmurowe (baza danych, serwer webowy, serwis), nie muszą być instalowane przez użytkowników systemu. Zostały one wdrożone przez nasz zespół i udostępnione w sieci publicznej poprzez API. Warunkiem ewentualnej instalacji wspomnianych modułów jest dostęp do aktywnej subskrypcji na platformie *Microsoft Azure* [1].

Baza danych Wdrożenie bazy danych odbywa się w dwóch krokach. Pierwszym z nich jest utworzenie pustej bazy danych w ramach usługi chmurowej *Azure Database for PostgreSQL* [2]. W ramach drugiego kroku odtworzyć należy kopię zapasową bazy danych w utworzonej bazie w chmurze. Jest to proces polegający na przesyłaniu danych z maszyny lokalnej do bazy o przydzielonym publicznym adresie IP przy pomocy narzędzia *pg_restore* [3]. Wspomniana kopia bazy danych dostępna jest w repozytorium [4].

Serwer webowy i serwis Kod źródłowy obu modułów jest dostępny w repozytorium [4]. Należy go pobrać i zbudować projekty *FreeturiloWebApi* oraz *NextBike ApiService* w konfiguracji *Release* [5] [6], po czym opublikować oba na platformie *Microsoft Azure* [7]. Najwygodniejszym do wykonania wymienionych kroków narzędziem jest *Visual Studio 2019*.

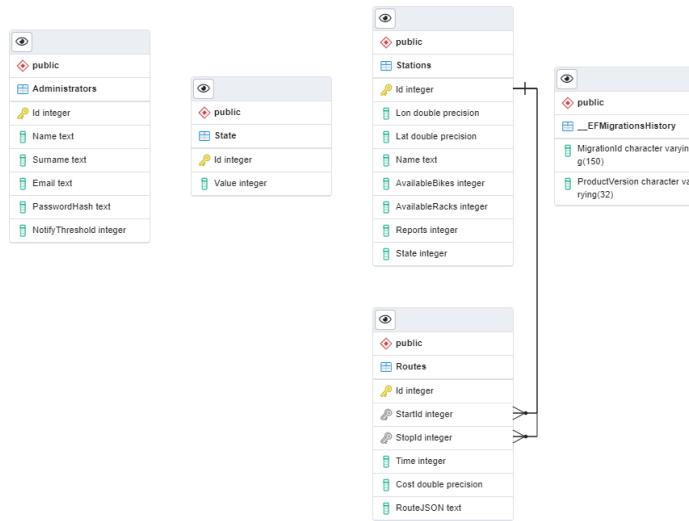
Aplikacja mobilna Jednym modułem, który musi zostać zainstalowany przez użytkownika w celu korzystania z systemu, jest aplikacja mobilna. Zarówno jej kod źródłowy, jak i plik *APK* dostępne są w repozytorium [8]. W celu instalacji aplikacji na urządzeniu z systemem *Android*, należy pobrać plik *APK* i umieścić go na urządzeniu, w katalogu dostępnym dla użytkownika (poprzez menadżer

plików). Następnie, korzystając z menadżera plików, wybrać należy pobrany plik APK. Jeżeli wersja systemu jest niższa niż 8.0, wymagane jest uprzednio zezwolić na instalację aplikacji z nieznanych źródeł w ustawieniach urządzenia. W przeciwnym wypadku, system uprzedzi użytkownika o potencjalnym zagrożeniu. Po zatwierdzeniu nastąpi instalacja pakietu.

5 Dokumentacja techniczna

Serwer Komunikacja z serwerem polega wyłącznie na wysyłanie requestów do publicznego API. Dokładna dokumentacja każdego endpointu jest częścią interfejsu i znajduje się pod odpowiednim adresem [9].

Baza danych Serwer bazy danych znajduje się pod odpowiednim adresem [10] i jest wystawiony na porcie 5432. Konfiguracja zezwala na łączenie się z bazą danych tylko usługom *Microsoft Azure* oraz prywatnym adresom IP członków zespołu. Dokładny schemat UML bazy danych jest przedstawiony na rysunku 1.



Rysunek 1: Schemat UML bazy danych

Serwis i biblioteka do deserializacji Użytkownik nie ma możliwości komunikacji z serwisem. Serwis komunikuje się z serwerem (poprzez REST API) oraz z interfejsem *NextBike*. Zwracana odpowiedź jest w formacie XML a jej dokładny schemat znajduje się w repozytorium z dokumentacją [11].

Aplikacja mobilna Dokumentacja kodu źródłowego aplikacji mobilnej została wygenerowana przy użyciu narzędzia *javadoc*. Znajduje się ona w całości w katalogu *javadoc* w repozytorium [8]. Aby przeczytać dokumentację, otworzyć należy plik *index.html* przy pomocy przeglądarki internetowej.

6 Instrukcja obsługi

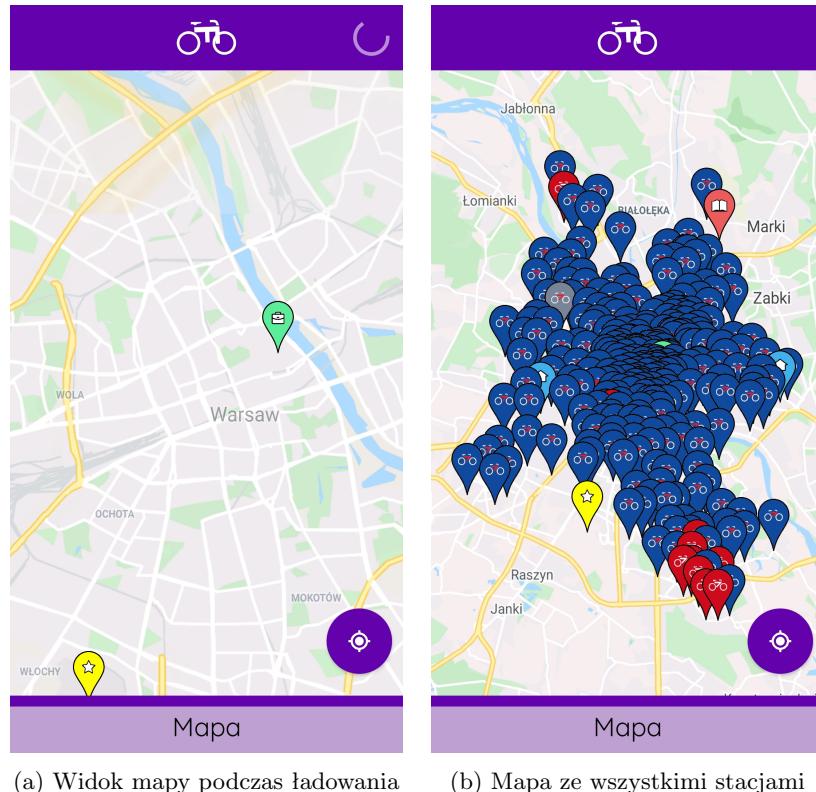
Po uruchomieniu aplikacji pojawia się widok główny (widoczny na rysunku 2), z którego możemy przejść do następnych widoków.



Rysunek 2: Menu główne

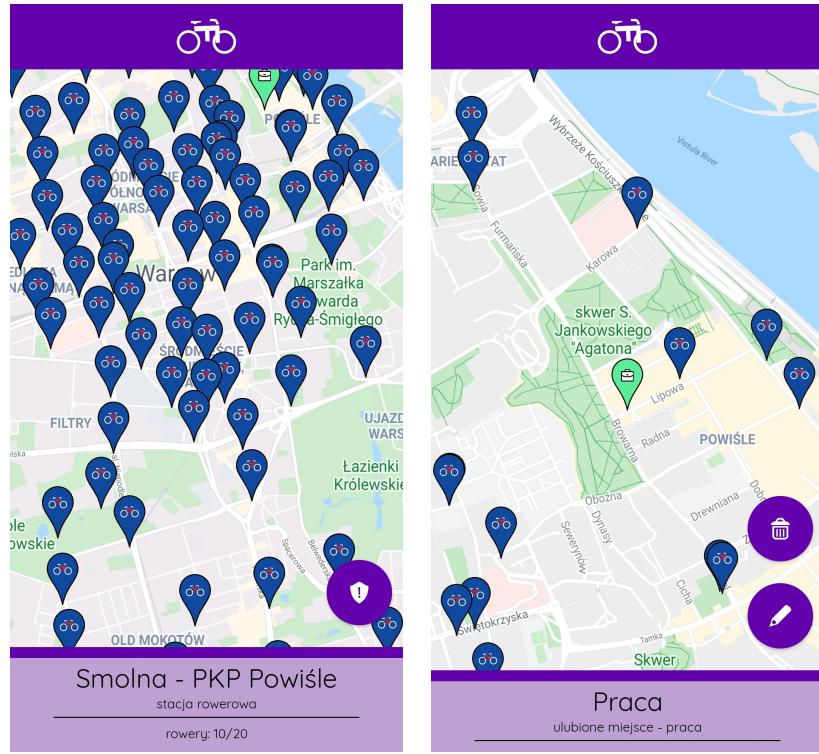
6.1 Mapa

Po wejściu do widoku mapy, na początku możemy w górnym prawym rogu zauważać widget oznaczający pobieranie danych z serwera (rysunek 3a). Po zakończeniu operacji na mapie pokazują się wszystkie stacje rowerowe oraz ulubione miejsca użytkownika (rysunek 3b).



Rysunek 3: Widok mapy

Po kliknięciu na dowolny znacznik na dole ekranu pojawia się belka ze szczegółami (rysunek 4a, 4b).



(a) Stacja rowerowa

(b) Ulubione miejsce użytkownika

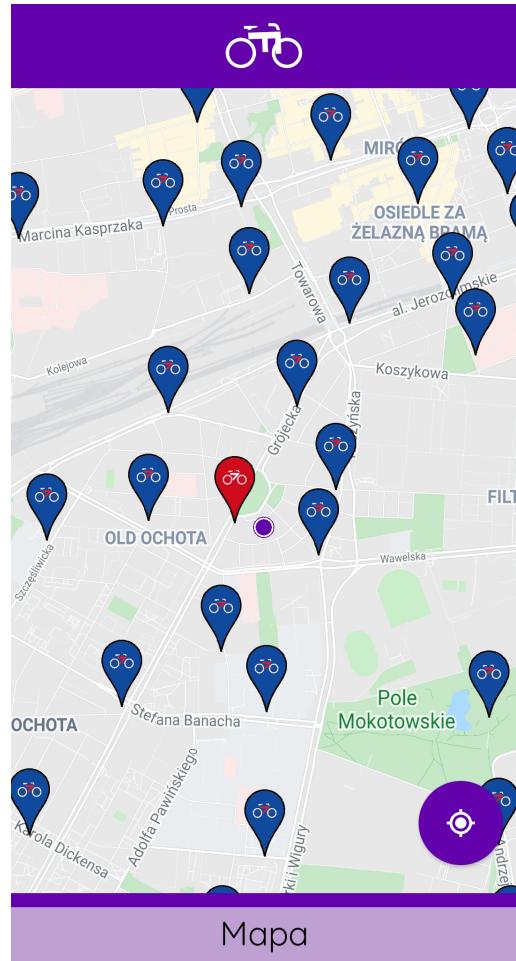
Rysunek 4: Szczegóły

Po kliknięciu na stację nad dolną belką po prawej stronie pojawia się przycisk służący do zgłoszenia stacji jako niesprawnej. Stacje zgłoszone jako niesprawne mają czerwone znaczniki. W przypadku ulubionego miejsca użytkownika, pojawiają się przyciski dodawania i edytowania informacji na temat ulubionego miejsca. Przytrzymanie punktu na mapie pozwala dodać nowe ulubione miejsce użytkownika. Widok dodawania oraz edycji jest przedstawiony na rysunku 5).



Rysunek 5: Widok dodawania oraz edycji ulubionego miejsca

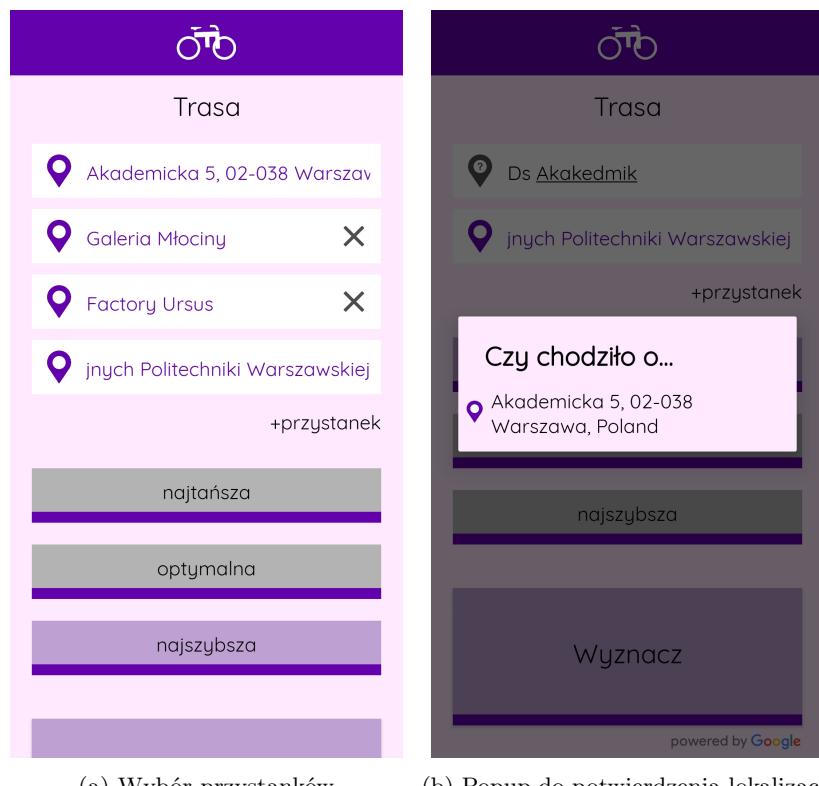
Mapa udostępnia również możliwość zaznaczanie lokalizacji użytkownika po uprzednim włączeniu funkcji lokalizacji na swoim urządzeniu mobilnym. Rysunek 6 przedstawia przykładową lokalizację.



Rysunek 6: Lokalizacja użytkownika na mapie

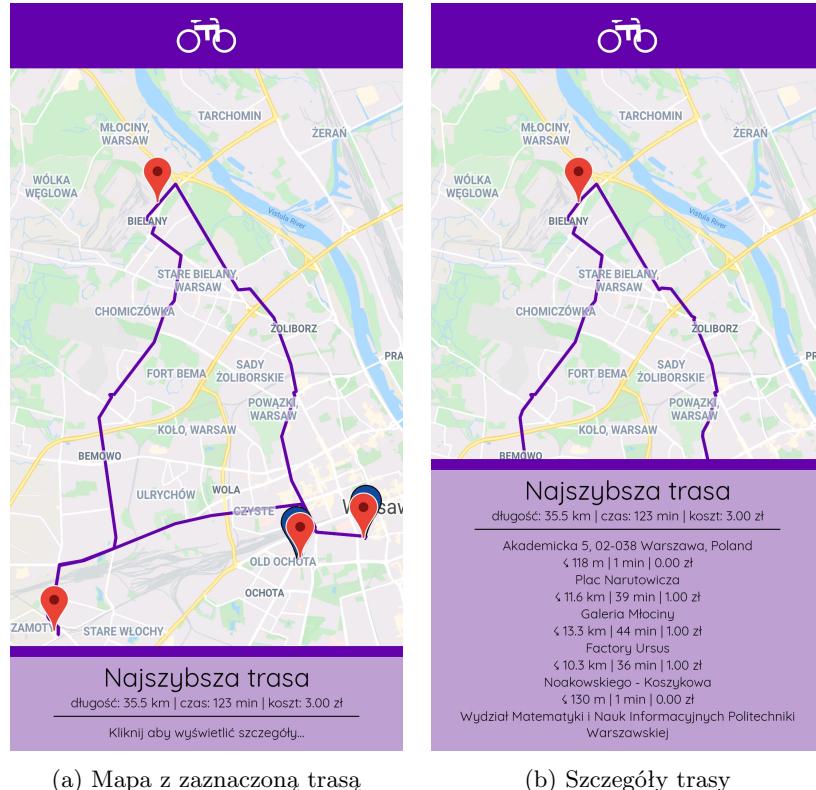
6.2 Trasa

W widoku trasy widoczne są pola do ręcznego wpisania stacji początkowej i końcowej. W trakcie wpisywania adresu aplikacja podpowiada lokalizacje. W przypadku, gdy użytkownik nie wybierze proponowanej lokalizacji z listy, przed wyznaczeniem trasy pojawi się prośba o wybranie jednej z wyświetlanych możliwości (rysunek 7b). Poniżej znajduje się również przycisk do dodania przystanków pośrednich (rysunek 7a). Dalej znajdują się trzy kryteria wyszukiwania trasy do wyboru oraz przycisk do wyznaczenia trasy.



Rysunek 7: Kryteria wyznaczenia trasy

Jeżeli formularz został wypełniony prawidłowo, aplikacja przechodzi do widoku mapy i przedstawia na niej wyznaczoną trasę. Czerwone znaczniki mają przystanki wpisane przez użytkownika, natomiast kolorem niebieskim oznaczone są stacje *Veturilo*. Po kliknięciu w dolną belkę rozwija się ona ukazując szczegóły trasy, m.in. koszt i czas każdego etapu (rysunek 8b).



Rysunek 8: Widok trasy na mapie

6.3 Historia

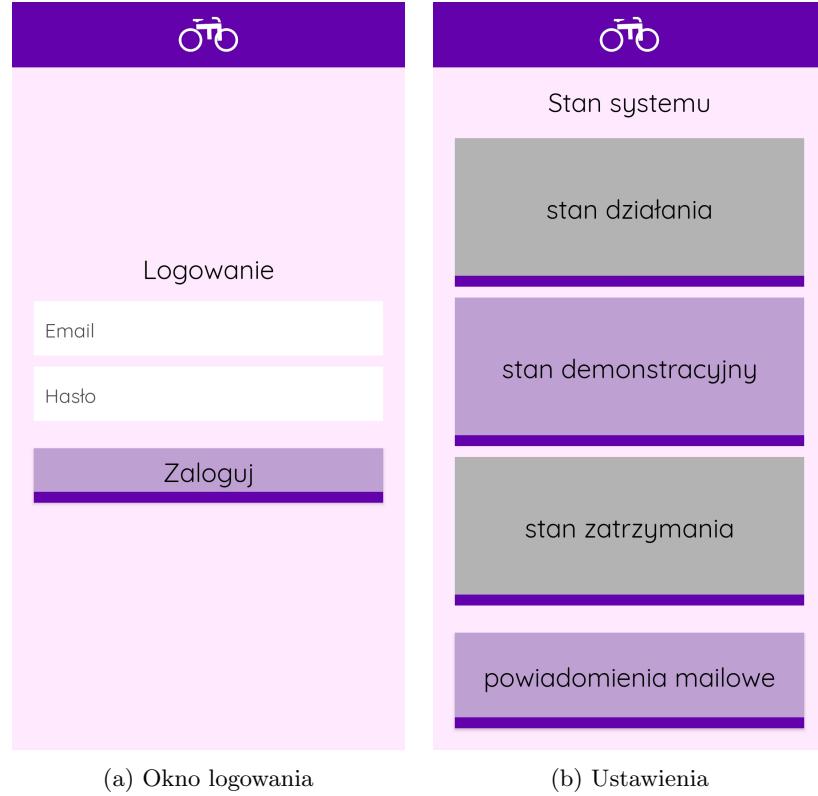
Z widoku głównego można przejść do historii wyznaczanych tras. Lista przedstawia ostatnio wyszukiwane trasy. Każdą z nich można usunąć poprzez kliknięcie na odpowiednią ikonę kosza. Po kliknięciu na trasę na liście zostanie ona ponownie wyznaczona i zaprezentowana użytkownikowi.



Rysunek 9: Widok historii tras

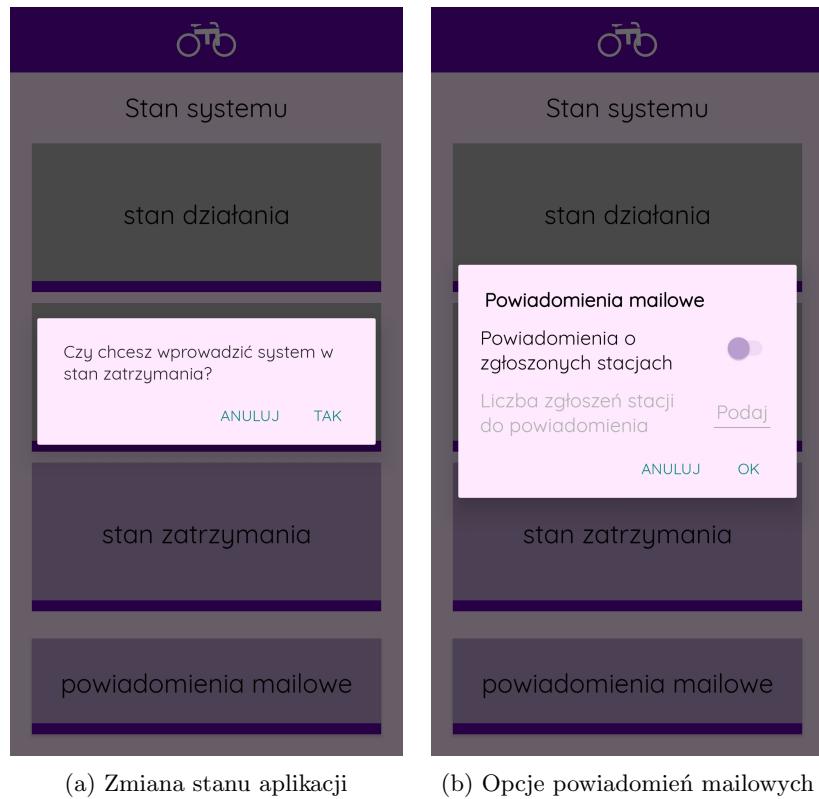
6.4 Panel administratora

W widoku głównym należy nacisnąć i przytrzymać logo aplikacji, aby przejść do okna logowania (rysunek 10a). Po udanej autoryzacji aplikacja przechodzi do widoku panelu administratora (rysunek 10b).



Rysunek 10: Panel administratora

Administrator z tego miejsca może wprowadzić system w jeden z trzech widocznych stanów: działania, administracyjny lub zatrzymania. Ma również możliwośćłączenia i wyłączenia powiadomień mailowych oraz ustawienia liczby zgłoszeń, po których mail ma zostać wysłany.



Rysunek 11: Akcje administratora

Literatura

- [1] Dostawca usług chmurowych Microsoft Azure
<https://azure.microsoft.com/pl-pl/>
- [2] Instrukcja korzystania z usługi Azure Database for PostgreSQL
<https://docs.microsoft.com/en-us/azure/postgresql/quickstart-create-server-database-portal>
- [3] Narzędzie do przywracania kopii zapasowych baz danych PostgreSQL
<https://www.postgresql.org/docs/11/app-pgrestore.html>
- [4] Repozytorium z kodem backendu
<https://github.com/Freeturilo/FreeturiloBackend>
- [5] Budowanie projektu w Visual Studio 2019
<https://docs.microsoft.com/en-us/visualstudio/ide/building-and-cleaning-projects-and-solutions-in-visual-studio?view=vs-2019>
- [6] Ustawianie konfiguracji Release w Visual Studio 2019
<https://docs.microsoft.com/en-us/visualstudio/ debugger/how-to-set-debug-and-release-configurations>
- [7] Publikowanie aplikacji ASP.NET w Azure App Service
<https://docs.microsoft.com/en-us/azure/app-service/quickstart-dotnetcore>
- [8] Repozytorium z kodem źródłowym aplikacji mobilnej Freeturilo
<https://github.com/Freeturilo/FreeturiloFrontend>
- [9] Dokumentacja interfejsu serwera webowego
<https://freeturilowebapi.azurewebsites.net/swagger/index.html>
- [10] Adres serwera bazodanowego
freeturilo-database.postgres.database.azure.com
- [11] Schemat XSD
<https://github.com/Freeturilo/FreeturiloDocumentation/blob/main/nextbike-parser/markers.svg>