



# Coldtime

aplikacja dla firmy POL-EKO

dokumentacja techniczna

# Spis treści

Coldtime – dlaczego?	3
Wybrane technologie	3
Narzędzia	4
Funkcjonalności aplikacji - ogółem	5
Sposób użycia aplikacji	7
Pierwsze uruchomienie	7
Dodawanie urządzenia	11
Ogólny interfejs	13
Widok karty urządzenia	14
Połączone	14
Niepołączone	14
„Nieodświeżone”	15
Widok urządzenia	16
Legenda:	17
Przewidywanie danych	19
Edycja oraz usuwanie urządzeń	25
Ustawienia	26
Obsługa wielu języków	28
Jak uruchomić aplikację na bazie kodu źródłowego lokalnie?	31
Dodatkowe pomysły na przyszły rozwój	32



# Coldtime – dlaczego?

Nazwą mojej aplikacji jest “Coldtime”. Celem zrobienia tego programu było zrealizowanie zadania konkursowego, które miało polegać na **stworzeniu aplikacji desktopowej dla firmy POL-EKO**, służącej do analizy danych pochodzących z lodówki SMART PRO.

Moim podstawowym zadaniem było to, aby aplikacja była dostępna do **jak największej liczby użytkowników** - dlatego Coldtime przetłumaczony jest na dwa języki, wybrane na podstawie docelowej grupy Państwa klientów - **język polski oraz angielski**. Ponadto aplikacja zapewnia obsługę **dla nieskończonej liczby urządzeń**, które są jednocześnie **bezproblemowo monitorowane**, z zachowaniem jak najlepszych praktyk. Kolejnym podstawowym konceptem dla mojego projektu była **prostota użycia** oraz jak najbardziej **przyjazny dla użytkownika oraz nowoczesny design**. Ponadto bardzo chciałem, aby projekt ten **czymś się wyróżniał**, był w jakiś sposób **innowacyjny** – dlatego postanowiłem wprowadzić prosty model **sztucznej inteligencji**, który przewiduje kolejne dane pomiarowe w czasie.

**Wszystkie podstawowe oraz dodatkowe założenia**, które zawarli Państwo w regulaminie konkursu, **zostały spełnione**, a dodatkowo sam od siebie dodałem parę funkcjonalności, które uważam, że idealnie spełnią potrzeby przyszłych użytkowników aplikacji.

## Wybrane technologie

- **Electron** - środowiska, które umożliwia tworzenie aplikacji desktopowych za pomocą technologii webowych – jest ono bardzo popularne i często używane, również wśród korporacji - przykładów powszechnie używanych aplikacji z użyciem tej technologii można wymienić niezliczoną ilość, jednak te najpopularniejsze to zdecydowanie: **Discord, Microsoft Teams, Visual Studio Code oraz Skype**.
- **TypeScript** – ze względu na to, żeby w przyszłości **kolejni programiści**, np. z Państwa firmy, **mogli szybko odnaleźć się w moim kodzie**. System typów znacznie ułatwia też zrozumienie kodu przez pryzmat przyszłych integracji z innymi serwisami.



- **SQLite** – ze względu na prostotę - **plikowa baza danych wydaje się najlepszym rozwiązaniem do takiego rodzaju aplikacji**; ponadto korzystam z ORM o nazwie **Prisma**
- **React** – ze względu na moje doświadczenie w tej technologii – oraz **Chakra UI** - bibliotekę komponentów
- **Tensorflow** – jedna z najpopularniejszych bibliotek do **uczenia maszynowego i sztucznej inteligencji** – mimo że kojarzona z Pythonem, ma ona też swoją implementację w JavaScriptie.
- **i18n wraz z i18next** - do internationalizacji - obsługa wielu języków
- **Dodatkowo:**
  - **Axios** – do pobierania danych przez REST API
  - **Recoil** – do zarządzania globalnym stanem aplikacji – osobiście uważam, że Redux nie byłby odpowiednim wyborem dla tego rodzaju aplikacji – wprowadziłoby to niepotrzebne skomplikowanie
  - **Apexcharts** – do wykresów - biblioteka ta nie jest bardzo popularna, jednakże osobiście uważam ją za najlepszą bibliotekę do wykresów w całym ekosystemie JavaScriptu.

Oczywiście, dla uruchomienia programu **nie jest potrzebna instalacja żadnych dodatkowych bibliotek** (tak, jak zostało to nakazane w regulaminie) – gotowy bundle zawiera wszystko, co potrzebne – nie trzeba nic doinstalowywać, a aplikacja powinna uruchomić się bezproblemowo nawet na czystej instalacji systemu – po prostu wystarczy dwa razy kliknąć na plik .exe.

## Narzędzia

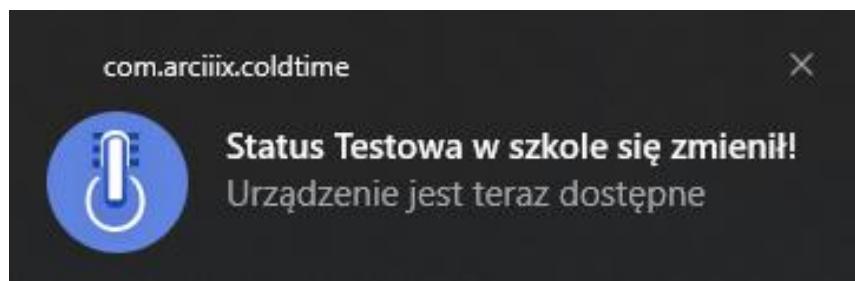
Oprócz tego użyłem narzędzi takich, jak:

- **GitHub** – do zarządzania historią wersji – aby projekt można było nazwać zrobionym profesjonalnie
- **Figma** - początkowo, do zamysłu/koncepcji na design
- **Notion** – do zarządzania listą zadań oraz pomysłami
- **Visual Studio Code** – do pisania kodu

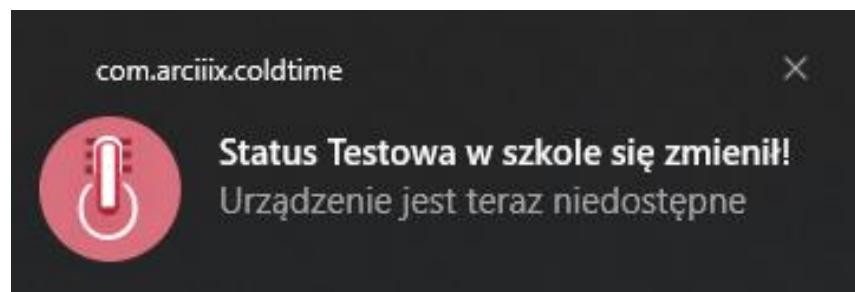


# Funkcjonalności aplikacji - ogółem

- **Dwujęzyczność**
- Obsługa **wielu urządzeń jednocześnie**
- **“Odkrywanie sieci”** – nie ma potrzeby podawać ręcznie adresu IP urządzenia
  - Coldtime samo wykryje włączone urządzenie w sieci
- Powiadomienia o tym, kiedy urządzenie **staje się dostępne / niedostępne**. Jeśli którykolwiek z dodanych urządzeń jest niedostępne, ikonka programu w pasku zadań zmienia kolor na czerwony.



Zrzut ekranu 1 - kiedy urządzenie przejdzie ze stanu "offline" do "online"

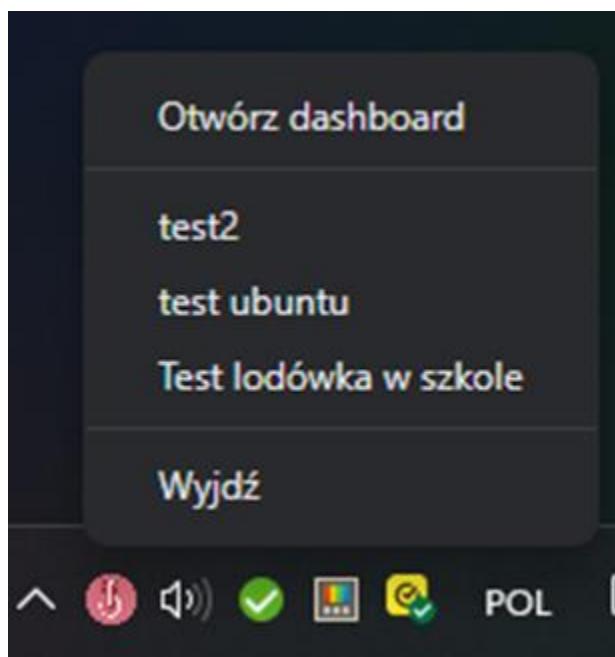


Zrzut ekranu 2 - kiedy urządzenie się rozłączy (np. jest niedostępne w sieci)

- **Dane historyczne** - z dowolnego okresu (do wyboru jest data początkowa oraz końcowa) + optymalizacja (w przypadku dużej ilości punktów są one kompresowane, a użytkownik w ustawieniach wybiera, ile maksymalnie punktów wykres może wyświetlać)
  - Wykres danych – z możliwością przybliżania, przesuwania itd.
  - Tabela z danymi, prezentująca je w sposób schludny i czytelny
- Drobne **statystyki dotyczące danych** - średnia temperatura danego dnia, czas zmiany statusu urządzenia (np. *włączono 2 minuty temu*)



- Eksport danych (zawartość dostosowuje się w zależności od języka aplikacji) do:
  - Microsoft Excel (.xlsx)
  - CSV
  - JSON
- **Możliwość dostosowania ustawień** - częstotliwości pobierania danych, ich zapisywania oraz maksymalnej ilości punktów na wykresie (do optymalizacji)
- Możliwość **pracy w tle** – domyślnie aplikacja tak naprawdę się nie zamyka, lecz pracuje w tle – aby ją ponownie otworzyć, należy nacisnąć na ikonkę na pasku zadań. Aby ją całkowicie wyłączyć, na pasku zadań, klikając prawym na ikonkę, należy wybrać “Wyjdź” lub “Quit”, w zależności od wybranego języka.



Zrzut ekranu 3 - Aplikacja podczas pracy w tle. Czerwona ikonka oznacza, że nie wszystkie urządzenia są podłączone.

- Możliwość edytowania oraz usuwania urządzenia. Ponadto każde urządzenie ma swoją nazwę.
- **Eksperymentalnie:** model sztucznej inteligencji, który przewiduje kolejne dane (temperaturę) w przyszłości



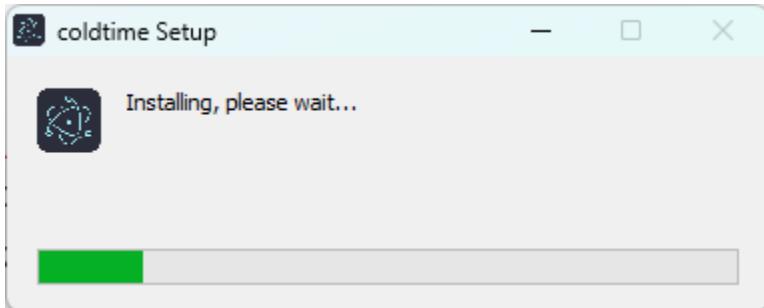
# Sposób użycia aplikacji

## Pierwsze uruchomienie

Aplikacja może zostać uruchomiona / zainstalowana na dwa sposoby:

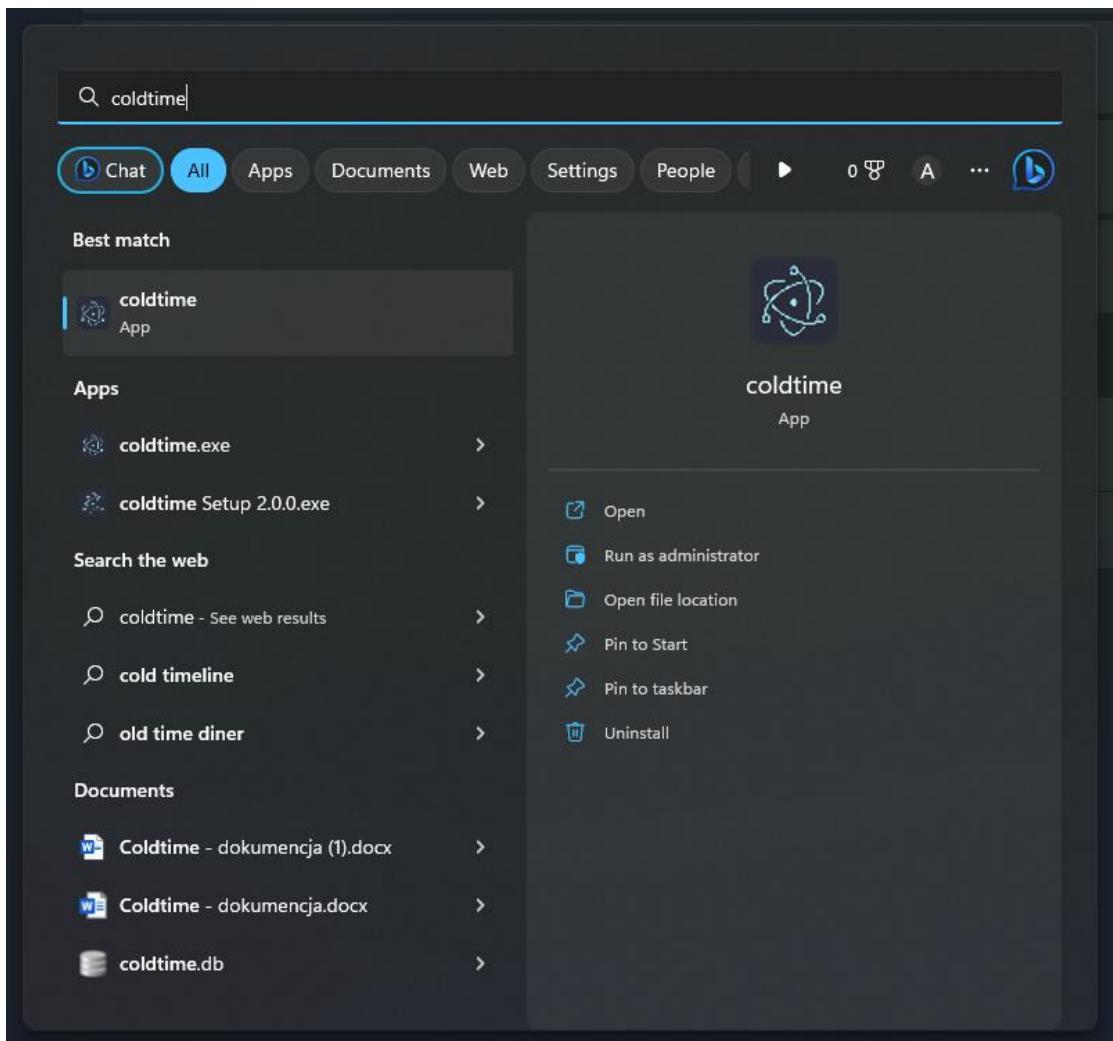
a. Przez instalator

a. Uruchamiamy plik z instalatorem (coldtime Setup 2.0.0.exe)



Po zainstalowaniu aplikacja powinna być dostępna bezpośrednio z poziomu systemu operacyjnego:





- b. Wersja portable (przenośna, bez instalacji, tylko uruchomienie)
- a. Uruchomiany plik coldtime.exe - aplikacja nie wymaga żadnych dodatkowych bibliotek, tak jak założono w zasadach konkursu



Coldtime – dokumentacja. Autor: [Artur Nowak](#)

win-unpacked				
Plik	Narzędzia główne	Udostępnianie	Widok	
← → ↑ ↓	Windows (C:) <> Użytkownicy > 2pr > Pobrane > finalne dist > dist > win-unpacked >	▼ 🔍	Przeszukaj: win-unpacked	?
Szybki dostęp	Nazwa	Data modyfikacji	Typ	Rozmiar
Szybki dostęp	locales	20.04.2023 17:53	Folder plików	
Pulpit	resources	20.04.2023 17:53	Folder plików	
Pobrane	chrome_100_percent.pak	20.04.2023 17:53	Plik PAK	127 KB
Dokumenty	chrome_200_percent.pak	20.04.2023 17:53	Plik PAK	176 KB
Obrazy	coldtime.exe	20.04.2023 17:53	Aplikacja	154 000 KB
20220920	d3dcompiler_47.dll	20.04.2023 17:53	Rozszerzenie aplik...	4 777 KB
20220930	ffmpeg.dll	20.04.2023 17:53	Rozszerzenie aplik...	2 703 KB
20221007	icudtl.dat	20.04.2023 17:53	Plik DAT	10 218 KB
20221104	libEGL.dll	20.04.2023 17:53	Rozszerzenie aplik...	473 KB
OneDrive	libGLESv2.dll	20.04.2023 17:53	Rozszerzenie aplik...	7 359 KB
Ten komputer	LICENSE.electron.txt	20.04.2023 17:53	Dokument tekstowy	2 KB
Sieć	LICENSES.chromium.html	20.04.2023 17:53	Microsoft Edge H...	6 651 KB
	resources.pak	20.04.2023 17:53	Plik PAK	5 249 KB
	snapshot_blob.bin	20.04.2023 17:53	Plik BIN	169 KB
	v8_context_snapshot.bin	20.04.2023 17:53	Plik BIN	472 KB
	vk_swiftshader.dll	20.04.2023 17:53	Rozszerzenie aplik...	5 014 KB
	vk_swiftshader_icd.json	20.04.2023 17:53	JSON Source File	1 KB
	vulkan-1.dll	20.04.2023 17:53	Rozszerzenie aplik...	894 KB

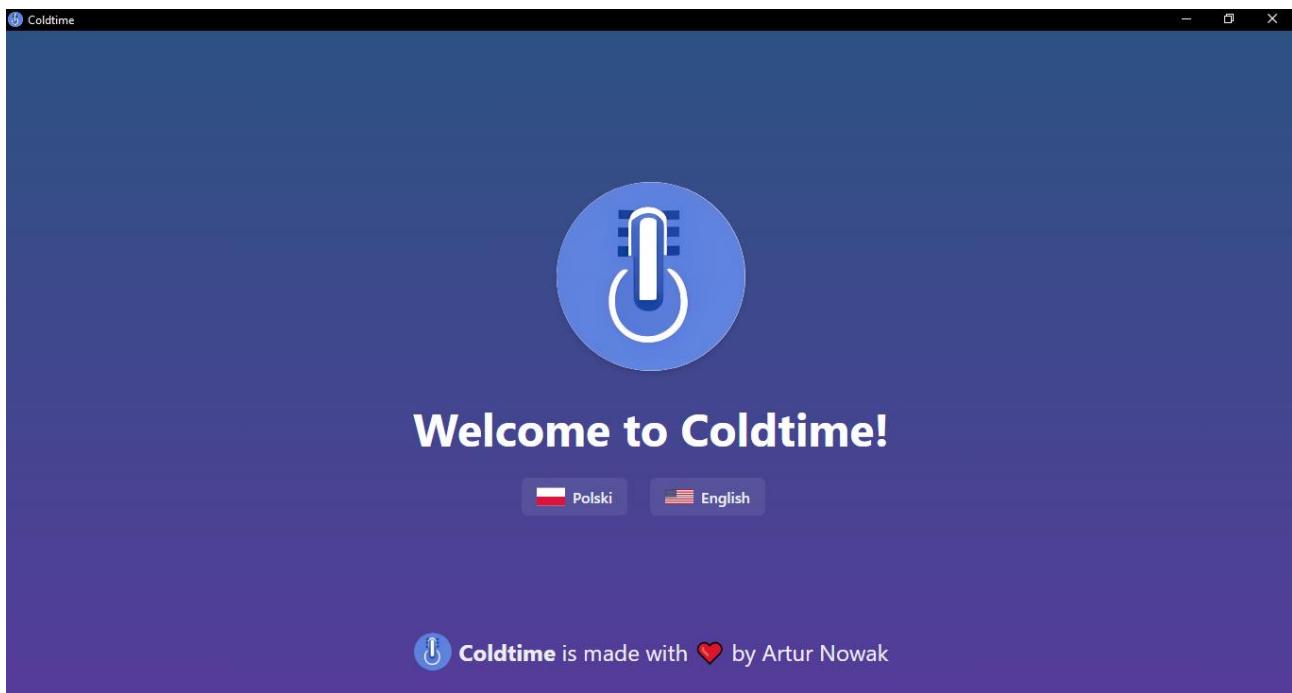
Elementy: 18



1. Po pierwszym uruchomieniu aplikacji, przy pustej bazie danych, naszym oczom ukąże się ekran:



Coldtime – dokumentacja. Autor: [Artur Nowak](#)

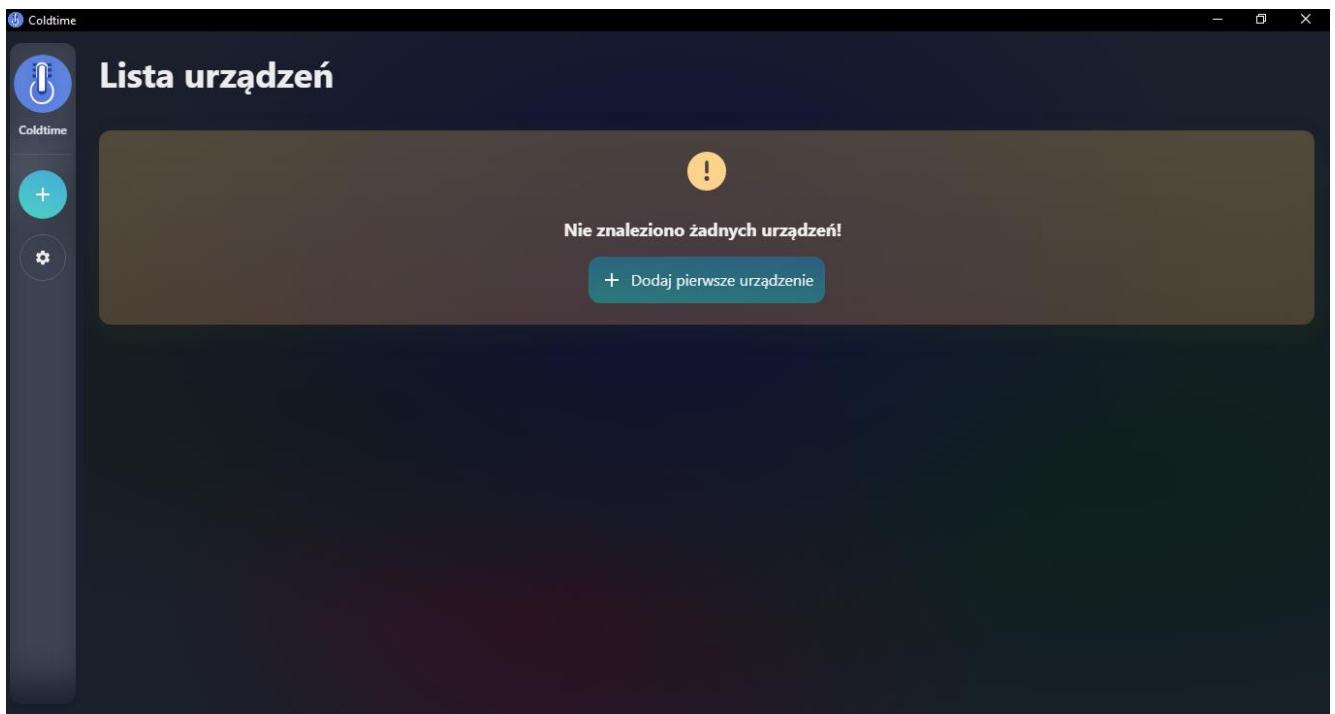


Zrzut ekranu 4 - Aplikacja przy pierwszym uruchomieniu. Na drugim zrzucie widzimy drugi etap animacji napisu początkowego - animacji zmiany języków.

2. Wybieramy jedną z opcji - decydują one o tym, w którym języku będzie wyświetlana nasza aplikacja – polskim lub angielskim.



Główny dashboard, po pierwszym uruchomieniu:



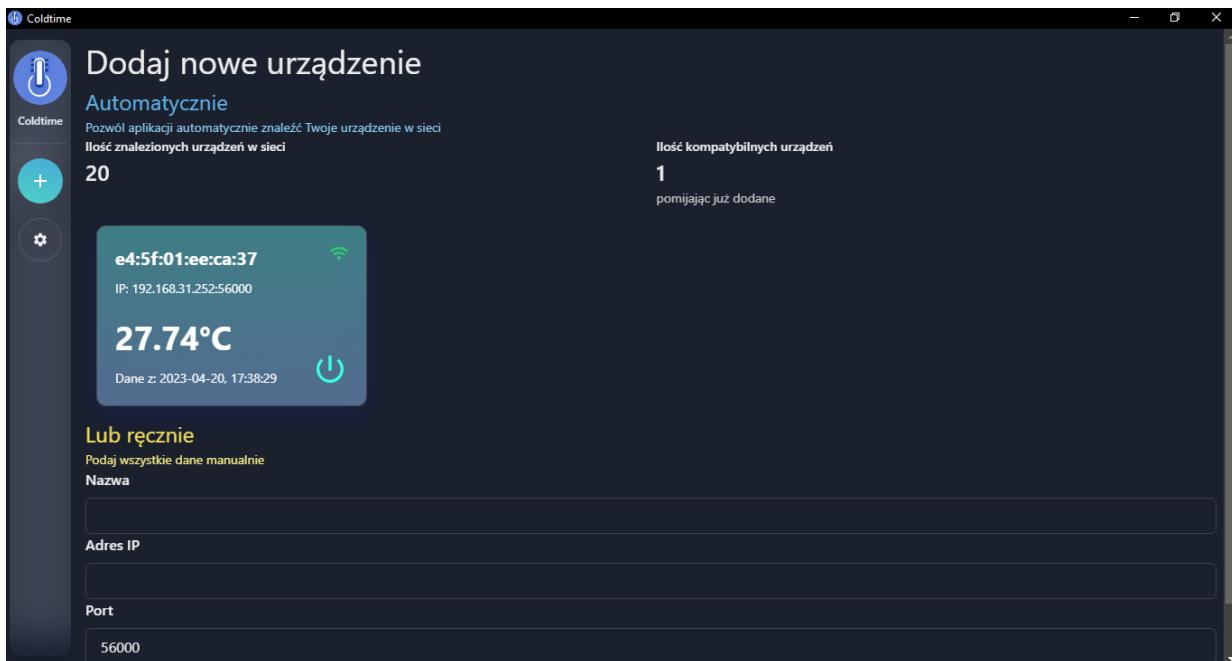
## Dodawanie urządzenia

Aby dodać pierwsze urządzenie, klikamy:

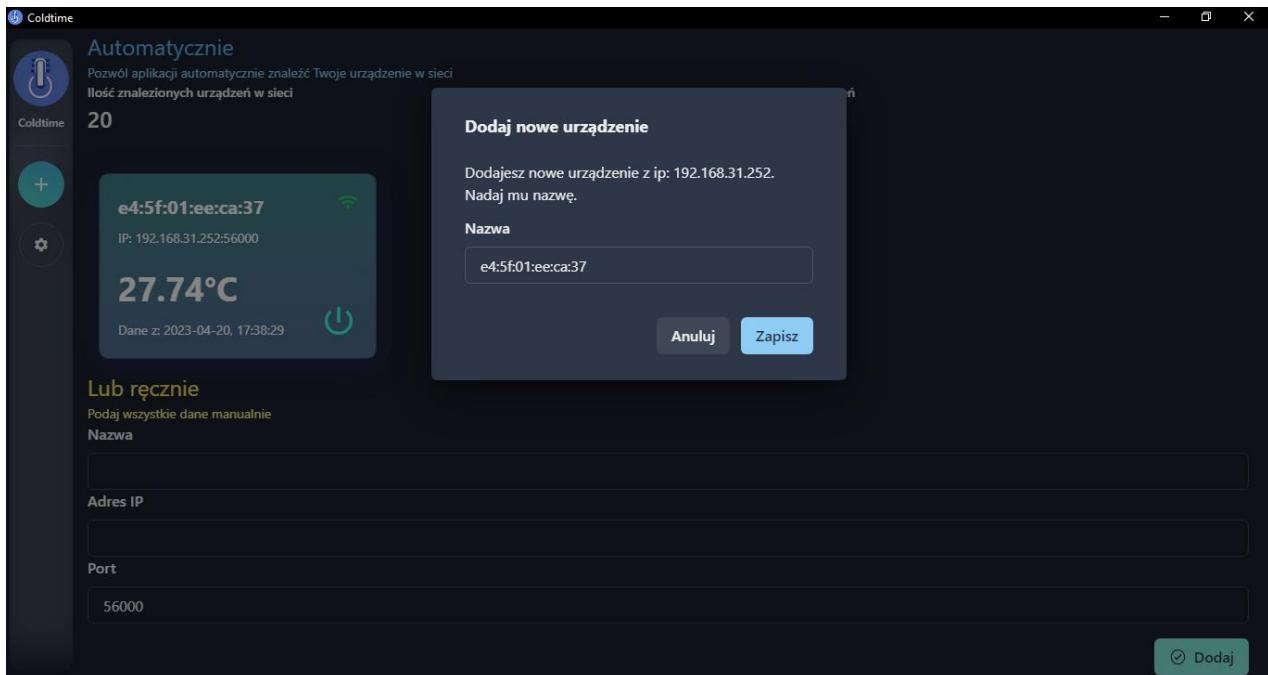
- Na przycisk "+" w sidebarze
- Na przycisk "Dodaj pierwsze urządzenie"

Jedną z wiodących funkcji mojej aplikacji jest automatyczne wykrywanie urządzeń w sieci (patrz: zrzut ekranu).





Urządzenie to zostało automatycznie wykryte. Aby je dodać, klikamy na nie, a następnie nazywamy:



Ewentualnie urządzenie można dodać ręcznie, poprzez wpisanie przyjaznej nazwy, adresu IP itd.



Lub ręcznie  
Podaj wszystkie dane manualnie

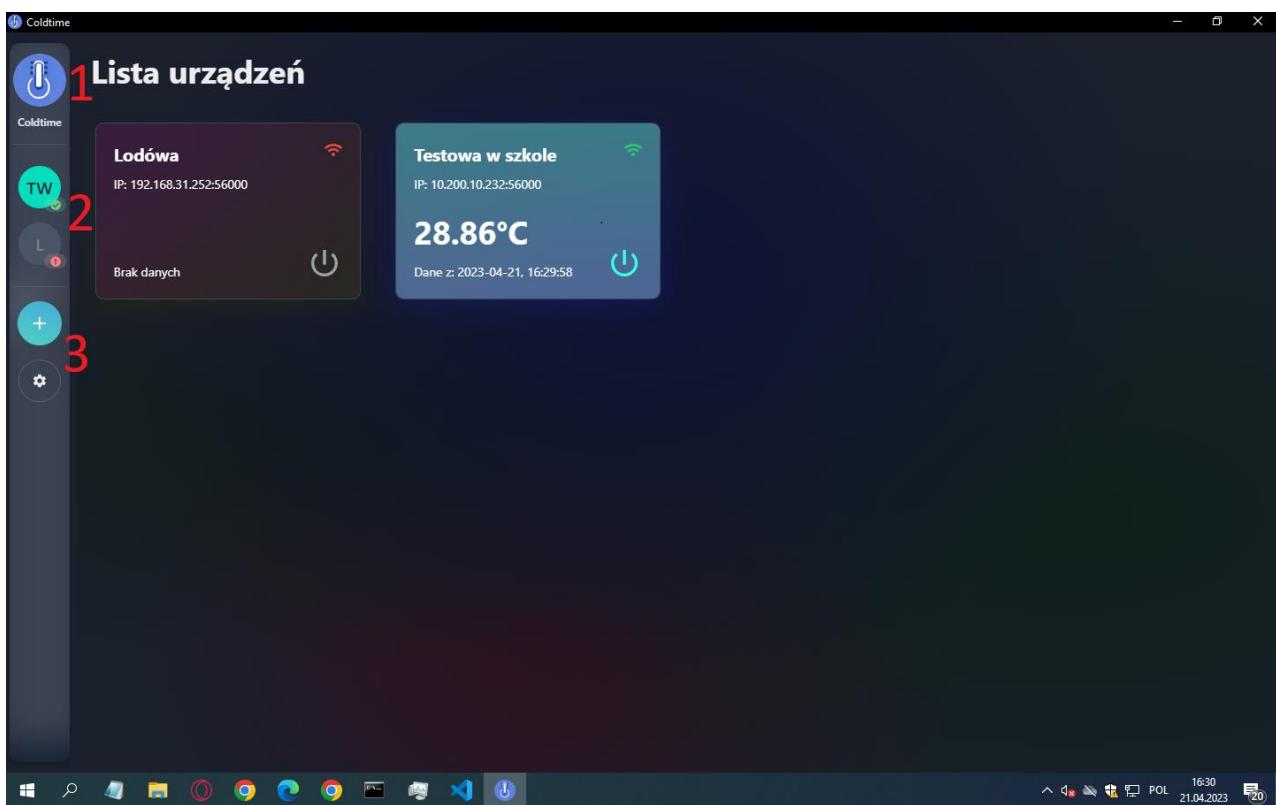
**Nazwa**

**Adres IP**

**Port**

Dodaj

## Ogólny interfejs



Po lewej stronie widzimy główną nawigację, sidebar – pierwszym jego elementem jest ikonka Coldtime (1), po której naciśnięciu aplikacja przenosi nas do strony głównej.

Później wymienione są kolejne urządzenia (2) wraz z ich obecnymi stanami.

Na dole (3) mamy przycisk do dodania urządzenia oraz ustawienia.



## Widok karty urządzenia

Połączone



1. Nazwa urządzenia
2. Adres IP oraz port
3. Temperatura
4. Czas pomiaru danych
5. Stan (czy włączona – *isRunning*)
6. Status połączenia

Niepołączone



# test ubuntu



IP: 192.168.0.156:3823

Brak danych



„Nieodświeżone”

tzn. stare dane zostały pobrane z bazy danych, a najnowsze dane są aktualnie pobierane w tle i za chwilę się pojawią:

# Testowa w szkole



IP: 10.200.10.232:56000

# 28.8°C



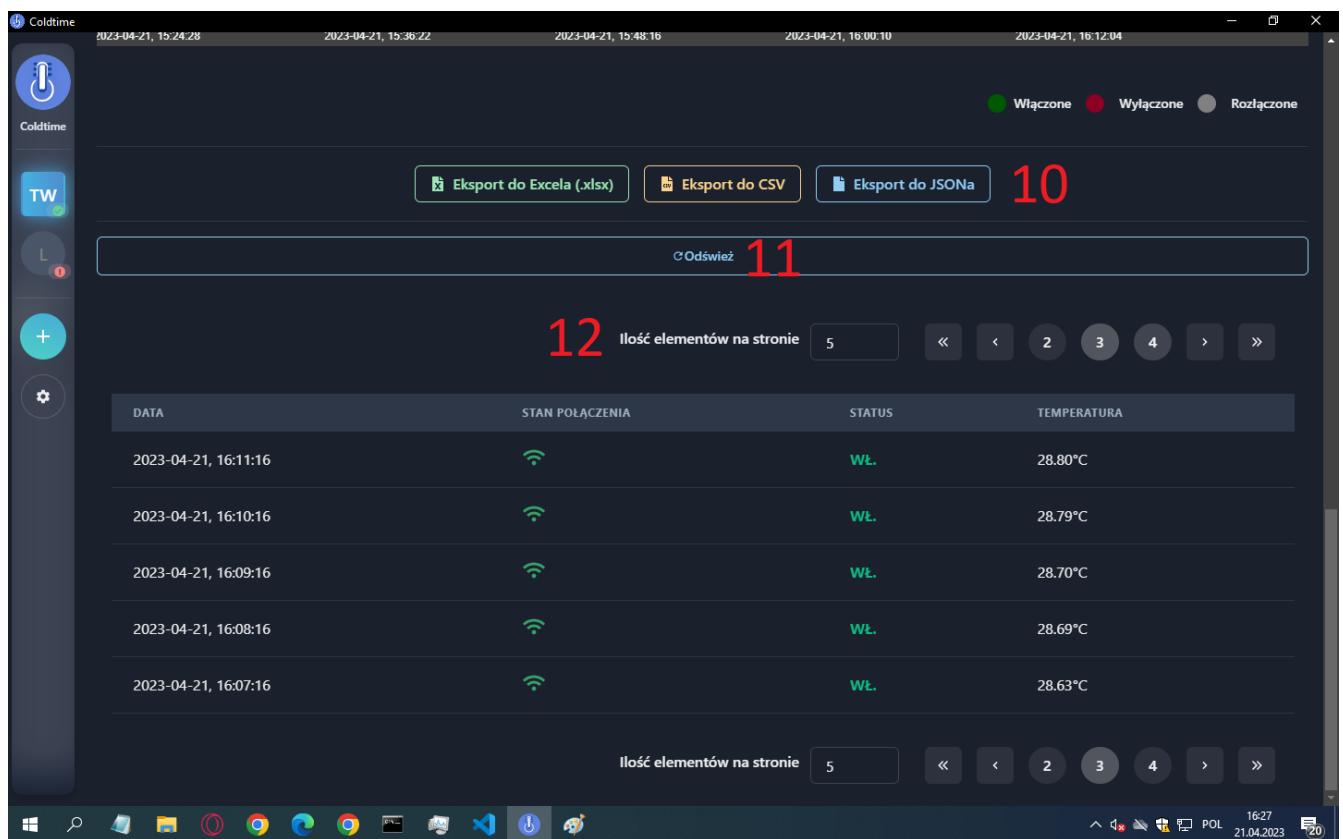
Dane z: 2023-04-21, 16:30:58



Coldtime – dokumentacja. Autor: [Artur Nowak](#)

## Widok urządzenia



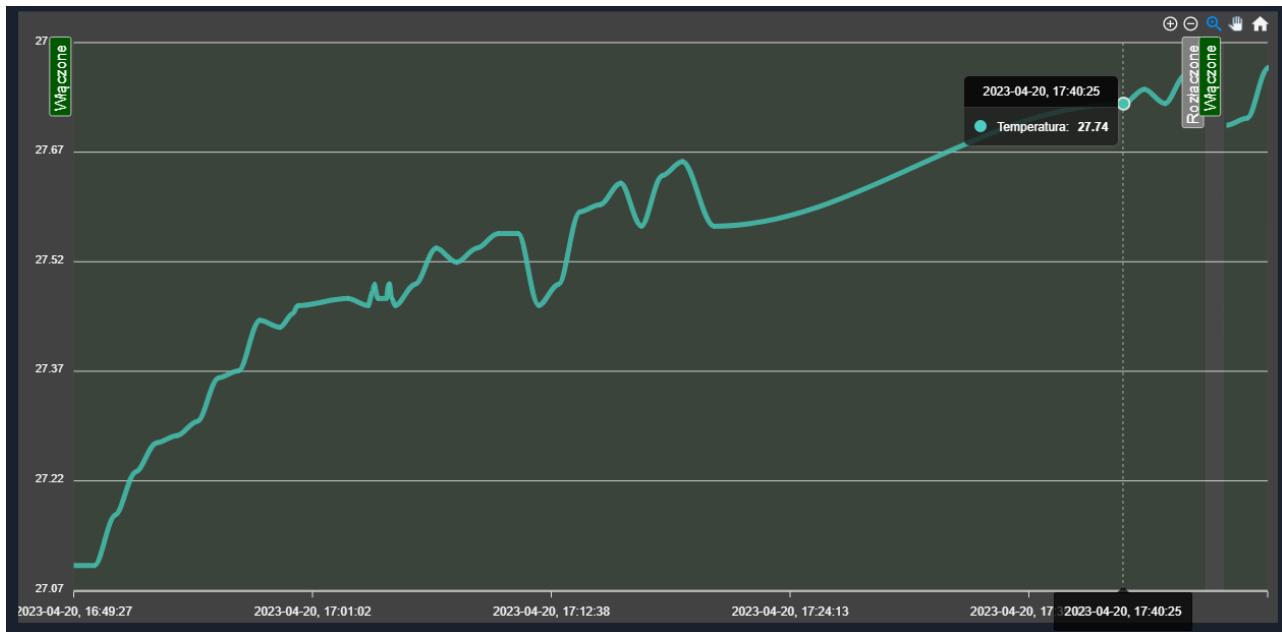


**Legenda:**



Coldtime – dokumentacja. Autor: [Artur Nowak](#)

1. Status połączenia z urządzeniem
2. Średnia temperatura – dziś (od 00:00 do 23:59)
3. Status urządzenia - włączone/wyłączone - oraz czas zmiany stanu, tzn. np. “od 2 dni” - 2 dni temu status się zmienił
4. Aktualna temperatura
5. Data i godzina ostatniej aktualizacji podanych danych oraz przycisk do odświeżania danych
6. Zakres czasowy danych historycznych – do wykresu. *Nie ustawiono daty* oznacza, że dane będą brane odpowiednio od zawsze do *daty końcowej* lub od *daty początkowej* do końca
7. Kolejna z wiodących funkcji mojej aplikacji – sztuczna inteligencja. Jest to jak na razie eksperymentalny feature. Jeśli mamy wystarczającą ilość danych, możemy przewidzieć kolejne pomiary.
8. Wykres historycznych danych urządzenia – z możliwością przybliżenia oraz oddalania, przesuwania itd. Zielone tło oraz adnotacja (na zdjęciu nr 8.1) oznaczają stan urządzenia, np.:



Zrzut ekranu 5 - szary obszar oznacza czas, w którym urządzenie było niedostępne

9. Legenda do wykresu – kolory obszarów

10. Funkcje eksportu – kolejno:

- a. Do Excela (.xlsx)
- b. Do CSV (.csv)



### c. Do JSONa (.json)

Przykładowy eksport do Excela wygląda następująco:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	czyPolaczono	data	temperatura	czyWlaczone											
2	PRAWDA	2023-04-20 17:40:25,139	27,74	PRAWDA											
3	PRAWDA	2023-04-20 17:20:31,076	27,57	PRAWDA											
4	PRAWDA	2023-04-20 17:18:59,797	27,66	PRAWDA											
5	PRAWDA	2023-04-20 17:17:59,789	27,64	PRAWDA											
6	PRAWDA	2023-04-20 17:16:59,782	27,57	PRAWDA											
7	PRAWDA	2023-04-20 17:15:59,770	27,63	PRAWDA											
8	PRAWDA	2023-04-20 17:14:59,758	27,6	PRAWDA											
9	PRAWDA	2023-04-20 17:13:59,746	27,59	PRAWDA											
10	PRAWDA	2023-04-20 17:12:59,736	27,49	PRAWDA											
11	PRAWDA	2023-04-20 17:12:00,000	27,46	PRAWDA											
12	PRAWDA	2023-04-20 17:11:01,946	27,56	PRAWDA											
13	PRAWDA	2023-04-20 17:10:01,940	27,56	PRAWDA											
14	PRAWDA	2023-04-20 17:09:01,912	27,54	PRAWDA											
15	PRAWDA	2023-04-20 17:08:01,899	27,52	PRAWDA											
16	PRAWDA	2023-04-20 17:07:01,888	27,54	PRAWDA											
17	PRAWDA	2023-04-20 17:06:01,885	27,49	PRAWDA											
18	PRAWDA	2023-04-20 17:05:02,152	27,46	PRAWDA											
19	PRAWDA	2023-04-20 17:04:54,285	27,47	PRAWDA											
20	PRAWDA	2023-04-20 17:04:53,249	27,47	PRAWDA											
21	PRAWDA	2023-04-20 17:04:52,058	27,47	PRAWDA											
22	PRAWDA	2023-04-20 17:04:47,217	27,49	PRAWDA											
23	PRAWDA	2023-04-20 17:04:45,696	27,49	PRAWDA											
24	PRAWDA	2023-04-20 17:04:43,295	27,49	PRAWDA											
	PRAWDA	2023-04-20 17:04:43,295	27,47	PRAWDA											

Zrzut ekranu 6 - z powodu wyboru języka polskiego nawet pola nagłówkowe są tłumaczone – tak samo jest w innych opcjach eksportu.

11. Przycisk do odświeżania danych
12. Tabela z historycznymi danymi (tymi samymi, jak na wykresie) – podzielona na strony, posiadająca funkcję dostosowania ilości elementów na pojedynczej stronie

## Przewidywanie danych

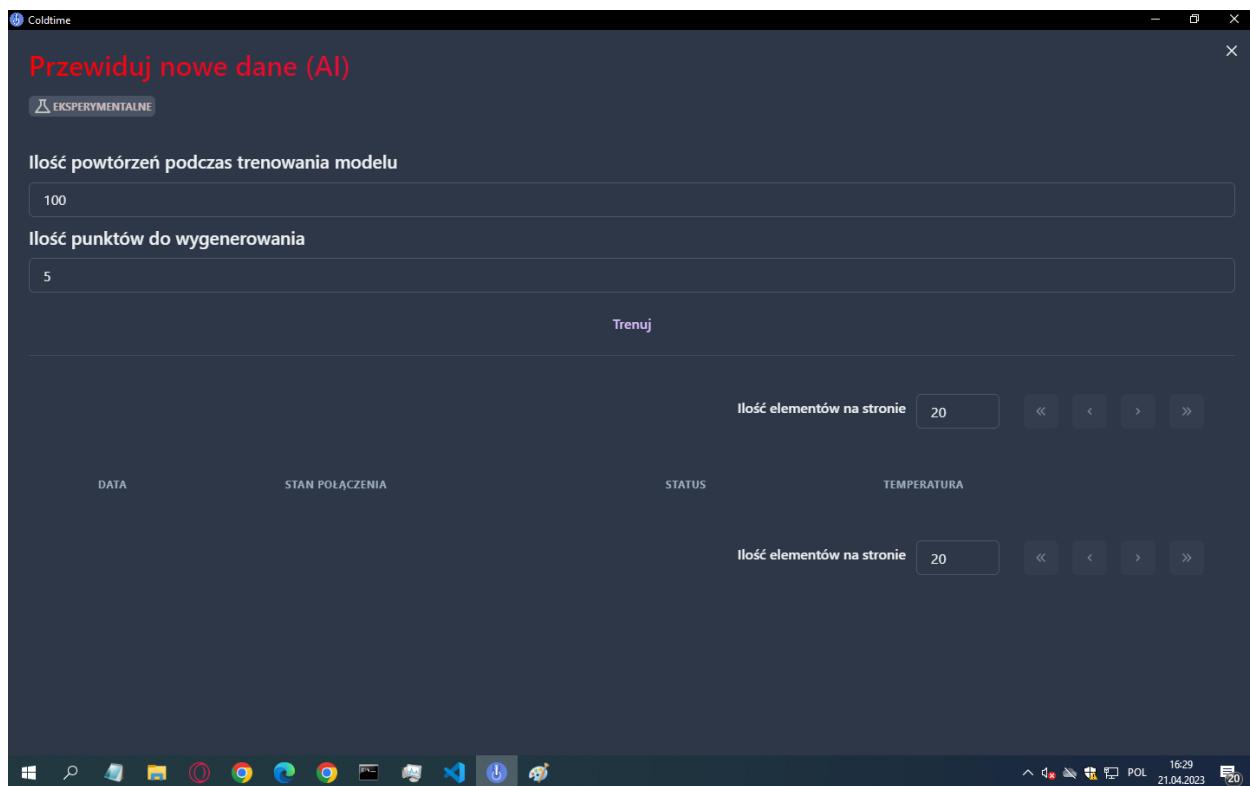
Dzięki współczesnej technologii z łatwością możemy stworzyć model sztucznej inteligencji, który postara się przewidzieć kolejne, przyszły pomiary, na podstawie tych obecnych. Takiego zadania podjąłem się też i ja.

Gdy mamy odpowiednią ilość danych (w przeciwnym przypadku przycisk nie będzie klikalny), na widoku szczegółu urządzenia klikamy:





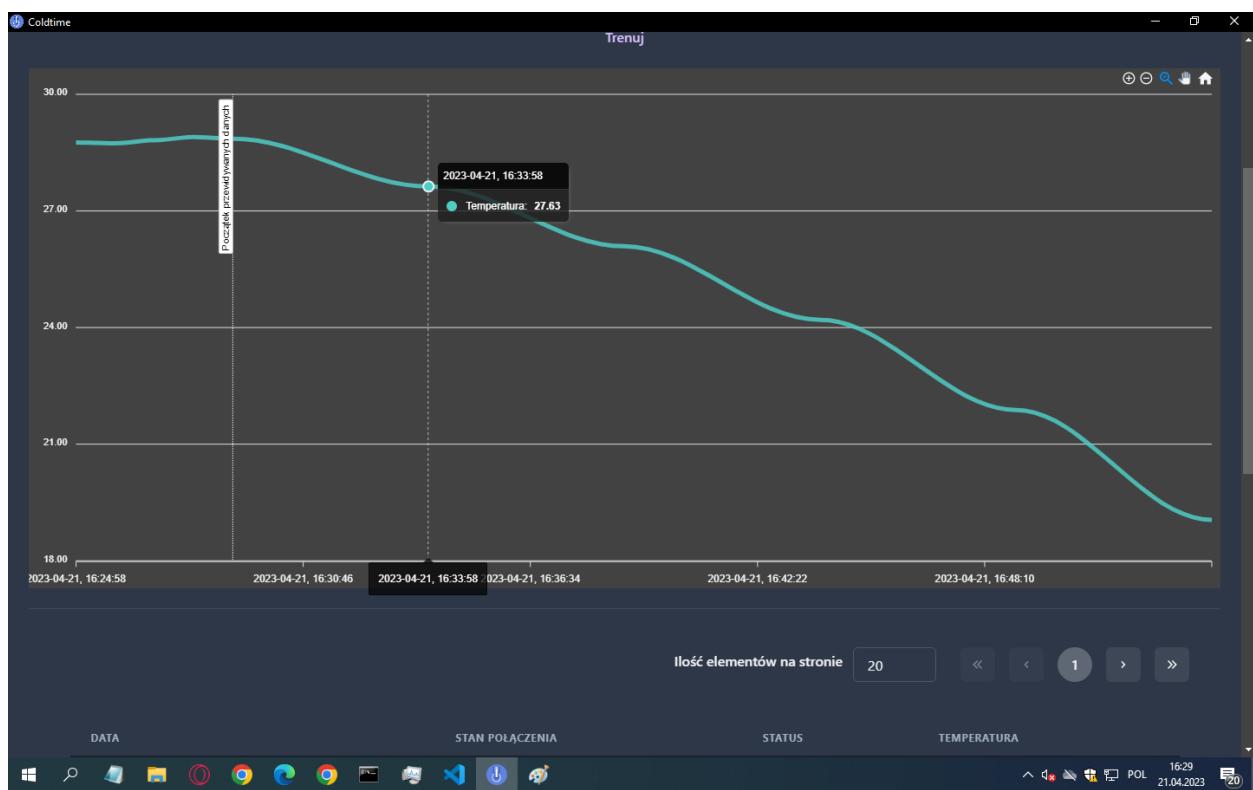
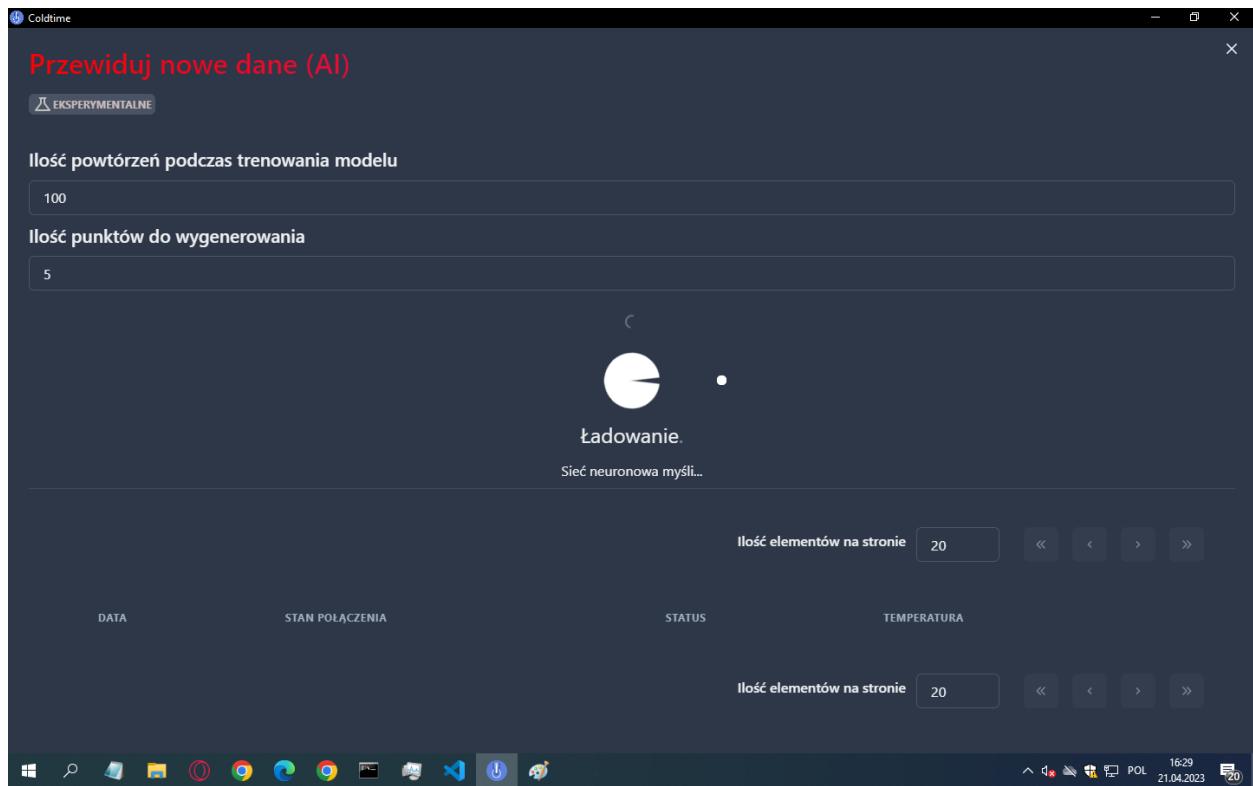
Wyświetli nam się ekran:



Wpisujemy ilość powtórzeń (ang. epochs) oraz ilość kolejnych punktów do wygenerowania. Punkty są generowane co 5 minut, a na wykresie zostanie ukazanych też kilka ostatnich historycznych punktów.

Po wpisaniu odpowiednich opcji (oczywiście są one walidowane, a więc nie możemy się pomylić) klikamy na przycisk „Trenuj”:





The screenshot shows a Windows desktop environment. At the top, there's a taskbar with various icons. In the center, a window titled "Coldtime" is open, showing a table of data. The table has four columns: "DATA", "STAN POŁĄCZENIA", "STATUS", and "TEMPERATURA". The data rows show timestamped measurements with a consistent "WŁ." status and varying temperatures from 19.05°C to 28.90°C. Below the table, the desktop background is visible, and at the bottom right, there's a system tray with icons and the date/time "21.04.2023 16:30".

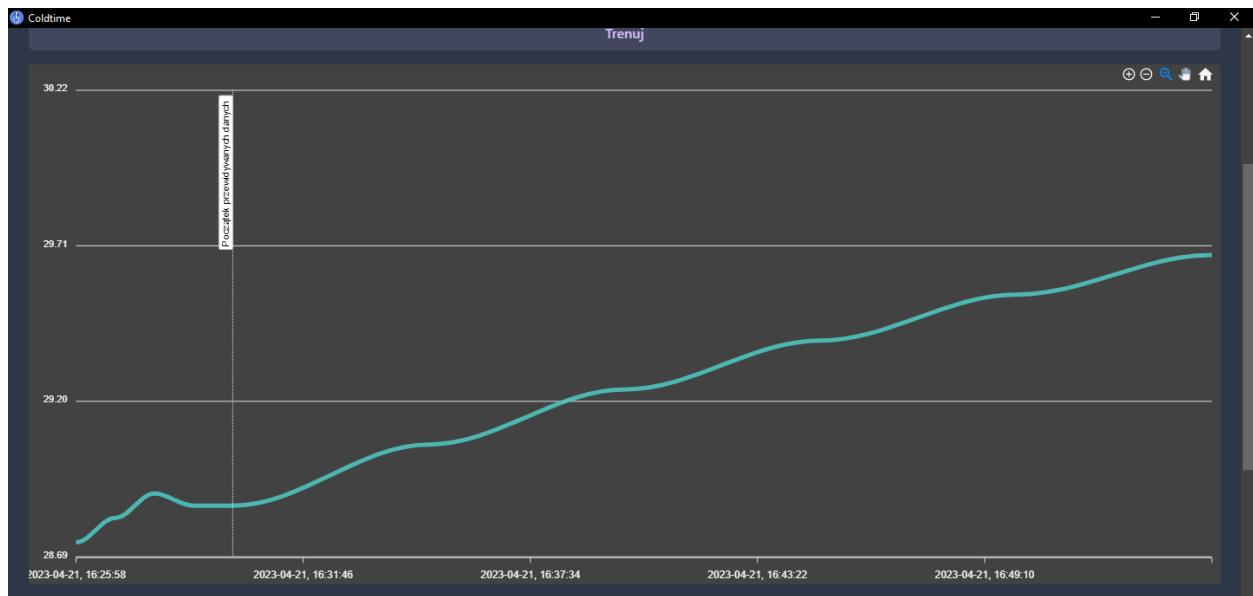
DATA	STAN POŁĄCZENIA	STATUS	TEMPERATURA
2023-04-21, 16:28:58	Wi-Fi	WŁ.	28.86°C
2023-04-21, 16:27:58	Wi-Fi	WŁ.	28.90°C
2023-04-21, 16:26:58	Wi-Fi	WŁ.	28.82°C
2023-04-21, 16:25:58	Wi-Fi	WŁ.	28.74°C
2023-04-21, 16:24:58	Wi-Fi	WŁ.	28.76°C
2023-04-21, 16:33:58	Wi-Fi	WŁ.	27.63°C
2023-04-21, 16:38:58	Wi-Fi	WŁ.	26.09°C
2023-04-21, 16:43:58	Wi-Fi	WŁ.	24.20°C
2023-04-21, 16:48:58	Wi-Fi	WŁ.	21.88°C
2023-04-21, 16:53:58	Wi-Fi	WŁ.	19.05°C

Jak widać, nasze AI wygenerowało nam potencjalne kolejne punkty, zarówno w formie wykresu, jak i tabeli.

Jak już wspomniałem, AI jest eksperymentalne, może przewidzieć nam kilka różnych wersji „przyszłej temperatury”, dlatego zalecana jest próba trenowania modelu kilka razy, aż do uzyskania satysfakcyjujących wyników.

Przykładowo, niektóre z potencjalnych scenariuszy, które wygenerowało w moim przypadku AI, wyglądają następująco:





Ilość elementów na stronie  1 < > >>

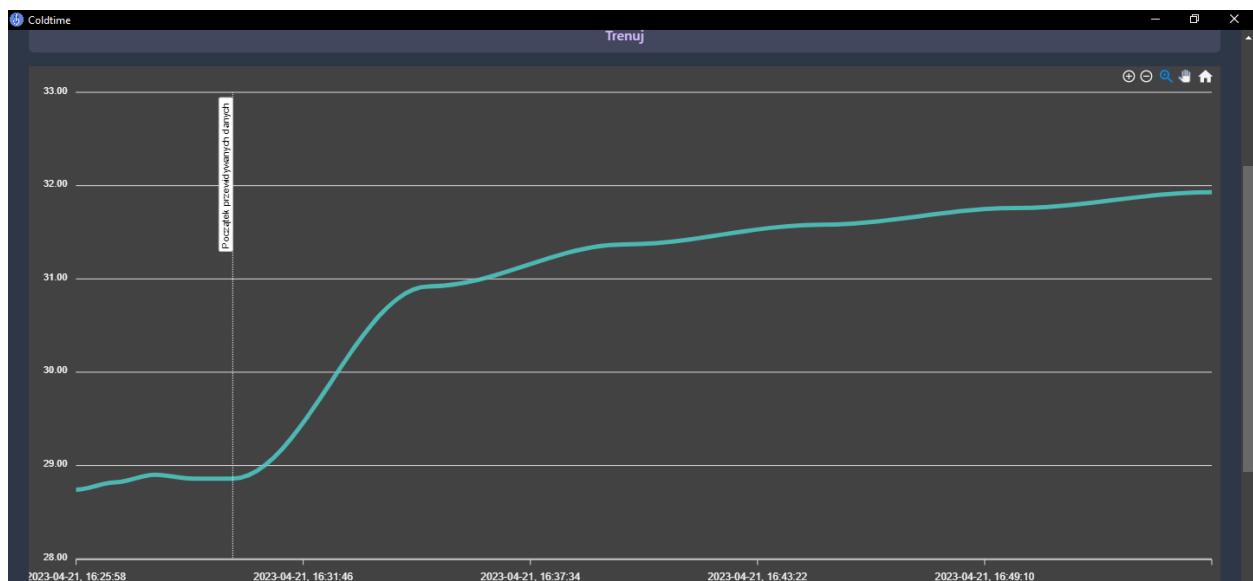
DATA

STAN POLĄCZENIA

STATUS

TEMPERATURA

16:30 21.04.2023 20



Ilość elementów na stronie  1 < > >>

DATA

STAN POLĄCZENIA

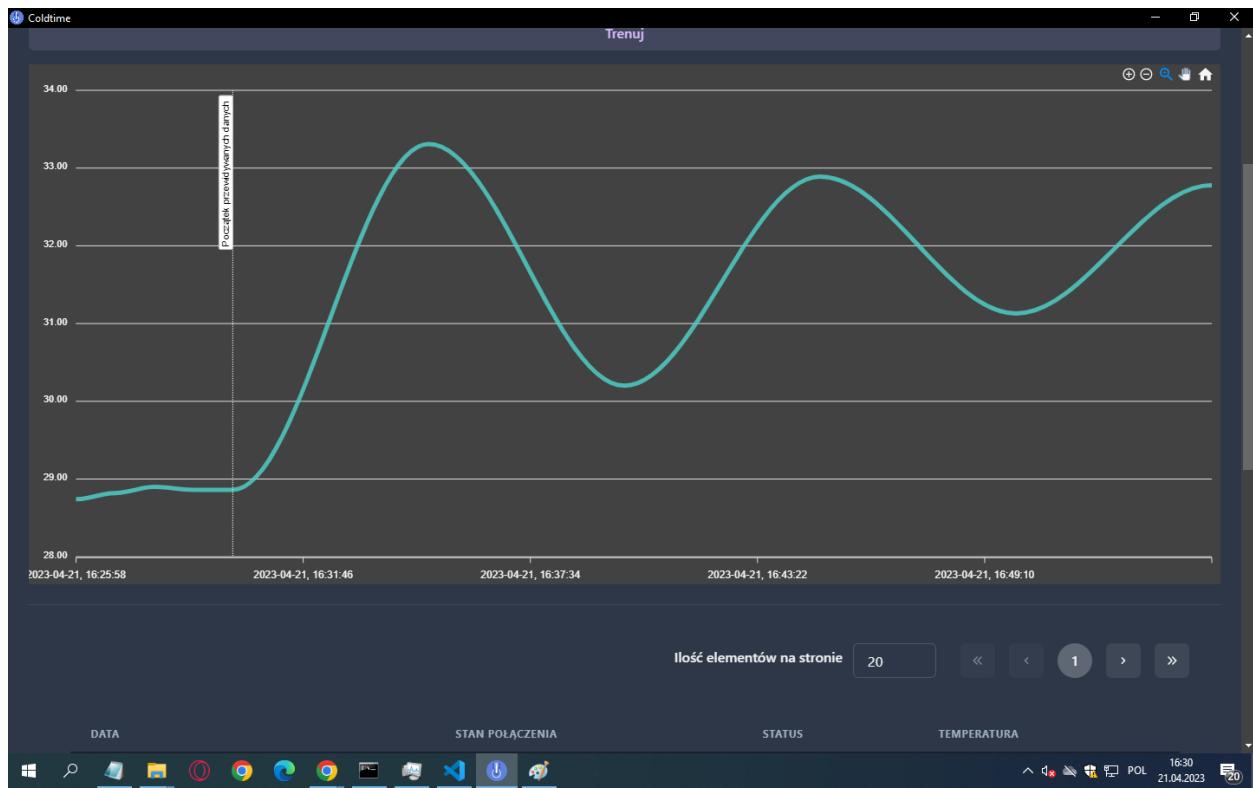
STATUS

TEMPERATURA

16:30 21.04.2023 20



Coldtime – dokumentacja. Autor: [Artur Nowak](#)

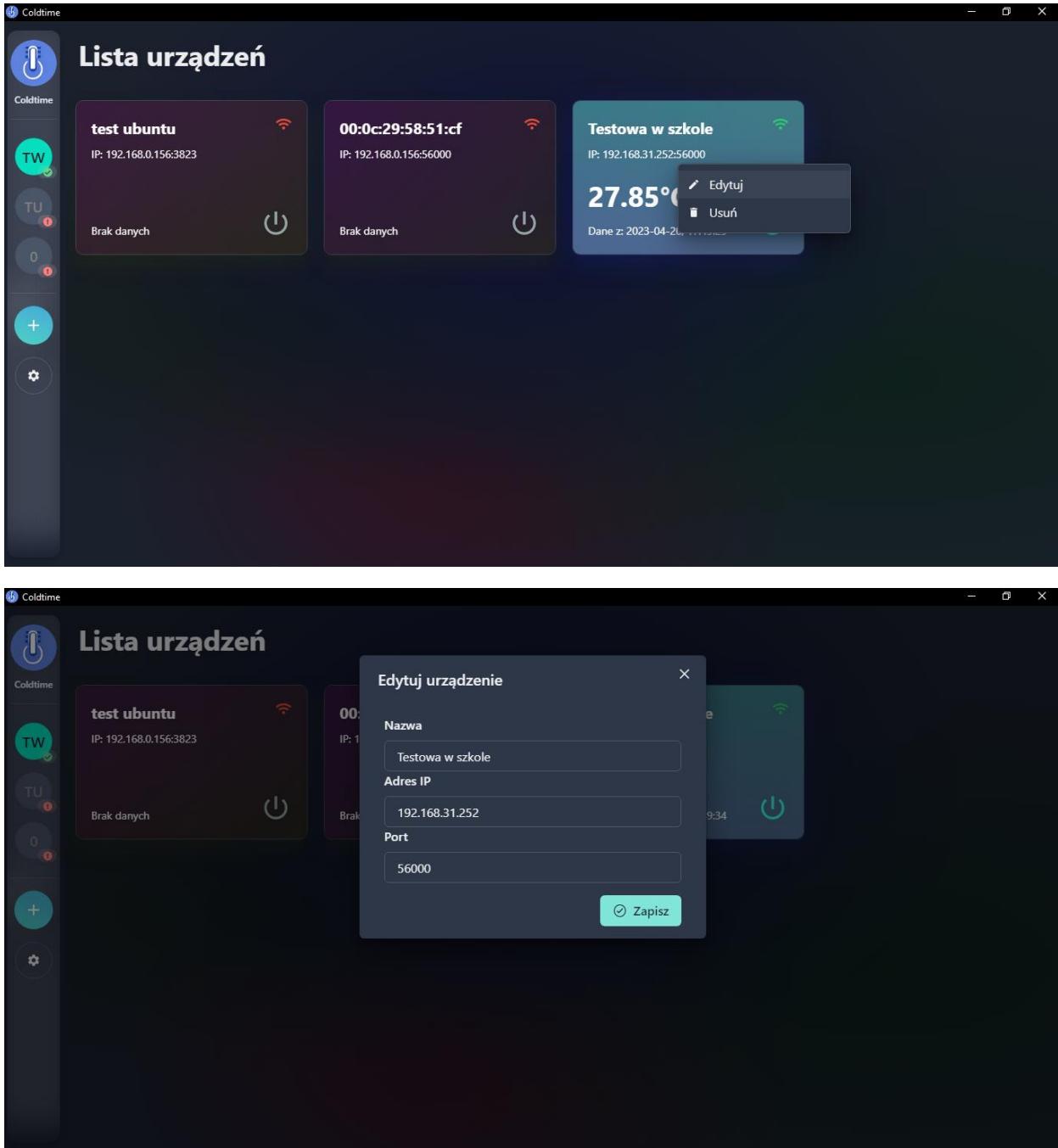


Mimo wszystko uważam stworzenie takowej funkcjonalności za osobiste osiągnięcie – temperatura, mimo że nie jest idealna, jest zbliżona do rzeczywistości. Dlatego też jest to jedna z najbardziej charakterystycznych funkcji Coldtime.



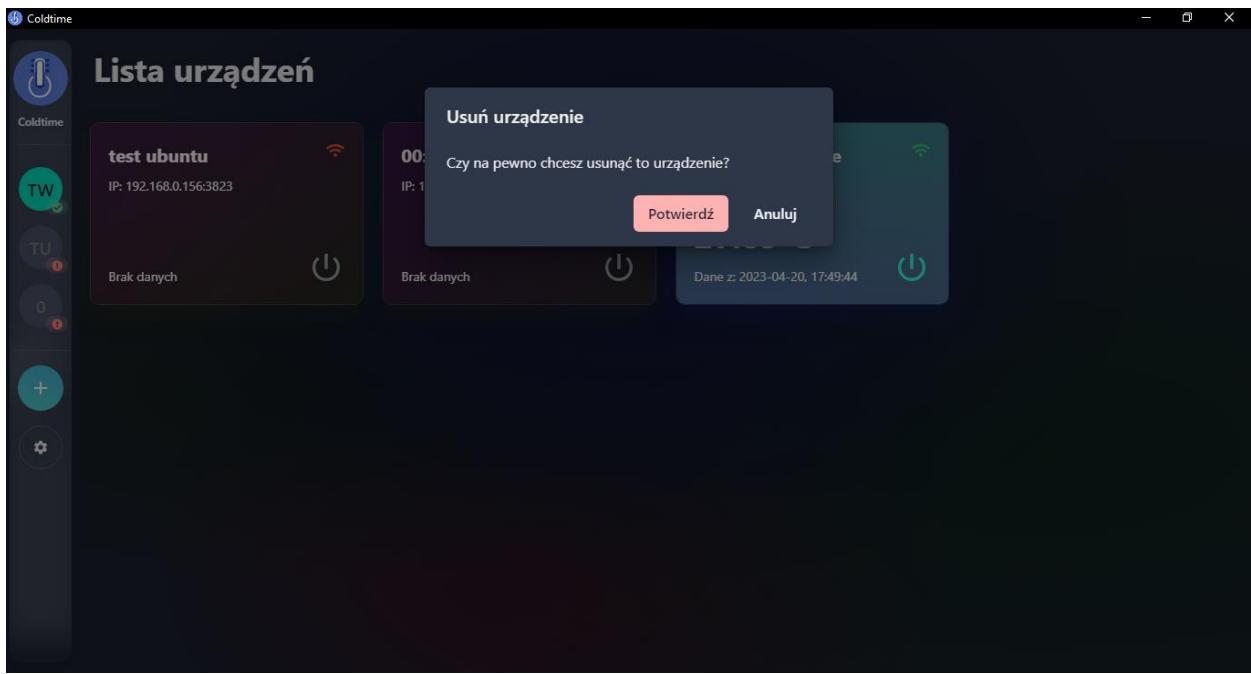
## Edycja oraz usuwanie urządzeń

Po kliknięciu prawym przyciskiem myszy na urządzenie na stronie głównej (dashboardzie), można wybrać dwie opcje:



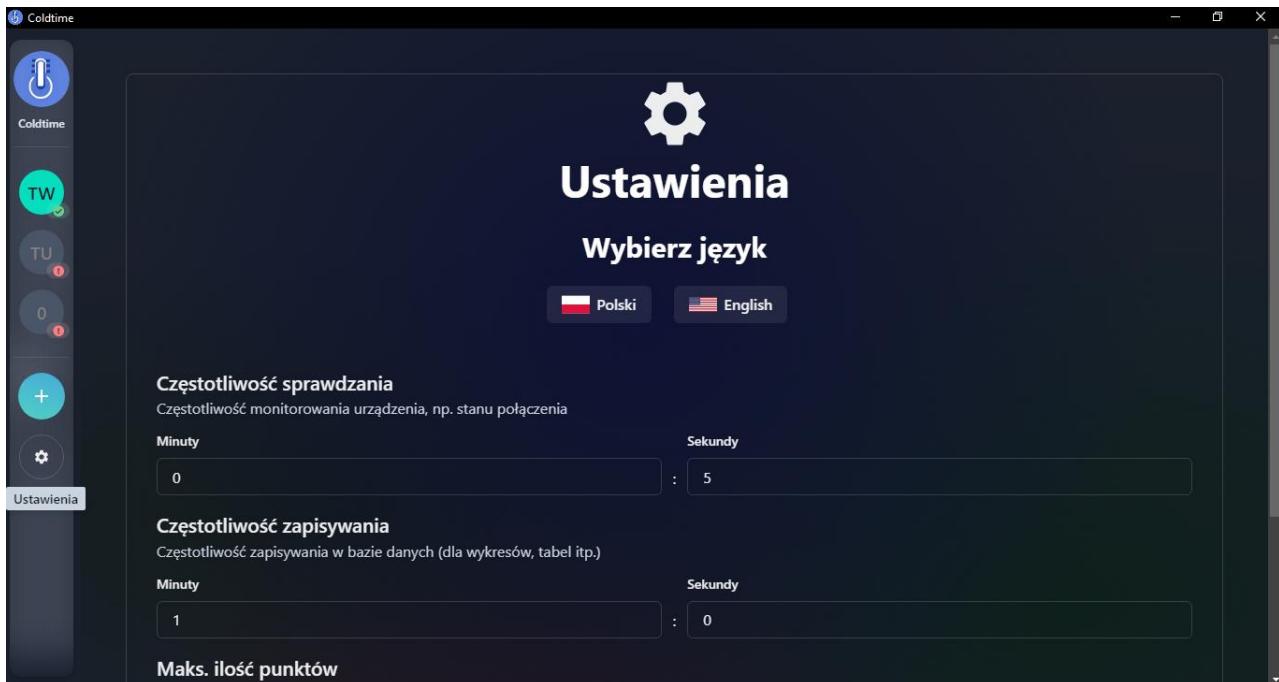
Zrzut ekranu 7 - Edycja urządzenia

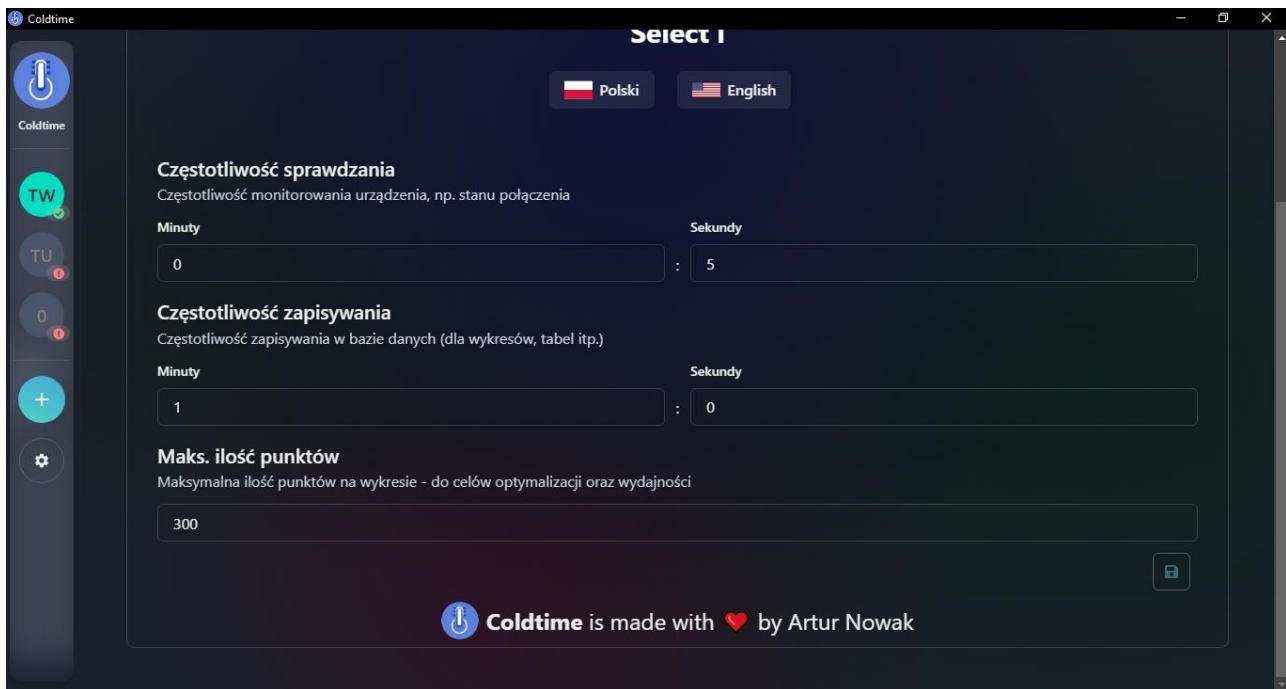




Zrzut ekranu 8 - usuwanie urządzenia

## Ustawienia

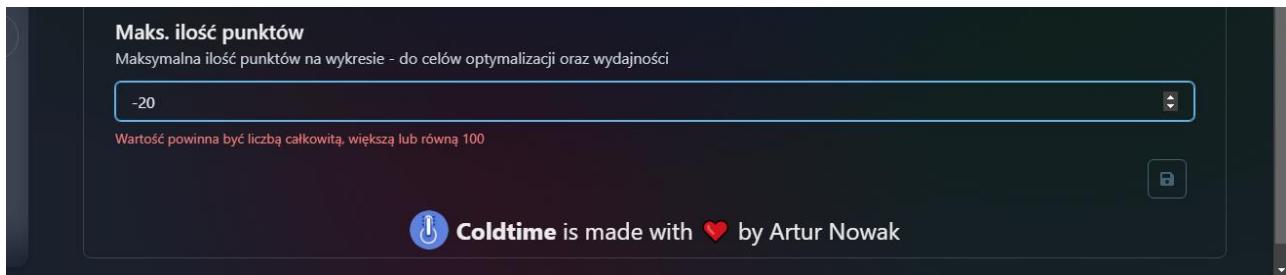




W ustawieniach możemy zmienić:

- Język
- Częstotliwość sprawdzania
- Częstotliwość zapisywania
- Maksymalną ilość punktów na wykresie

Wszystkie te opcje są dokładnie opisane oraz walidowane:

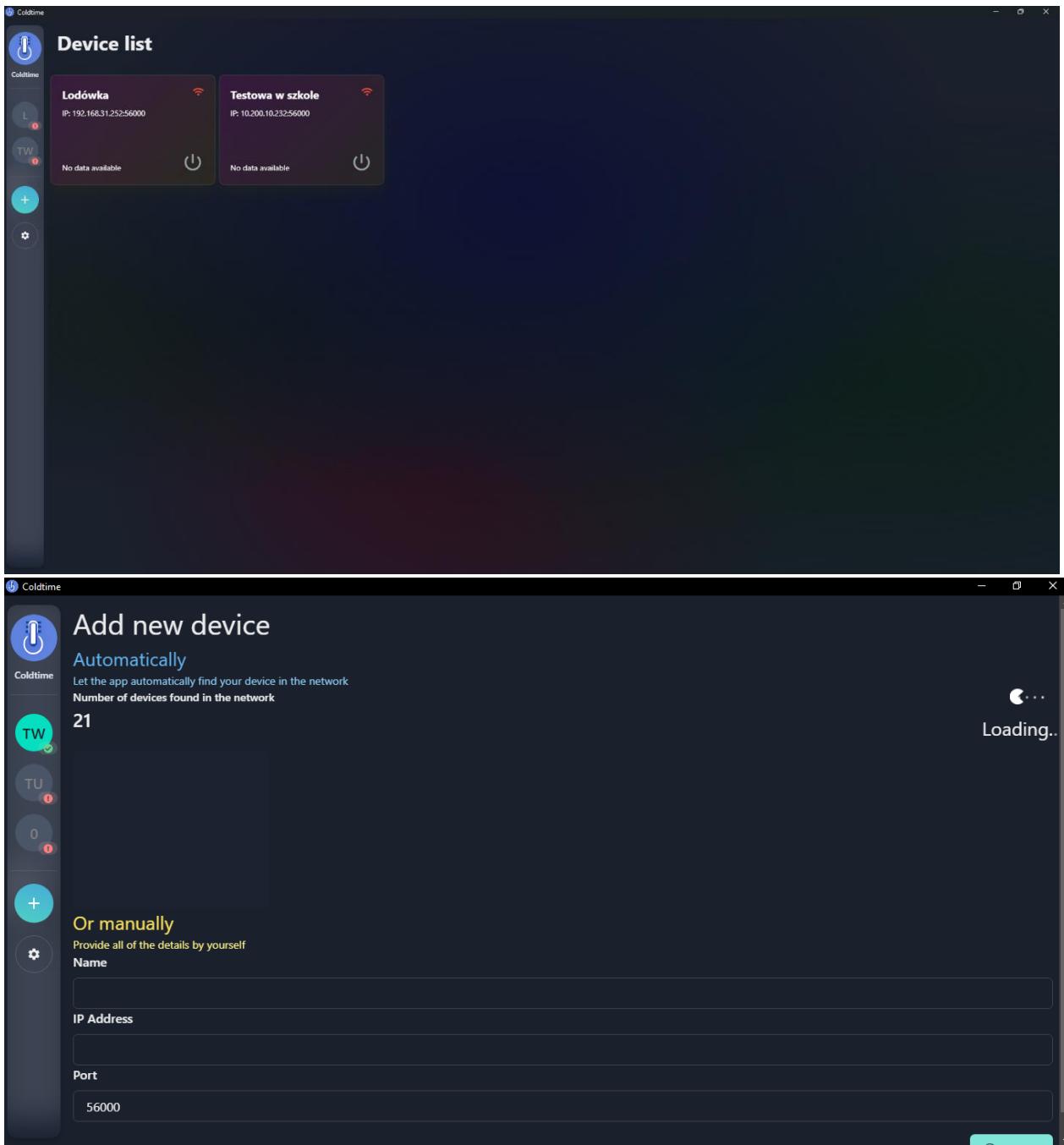


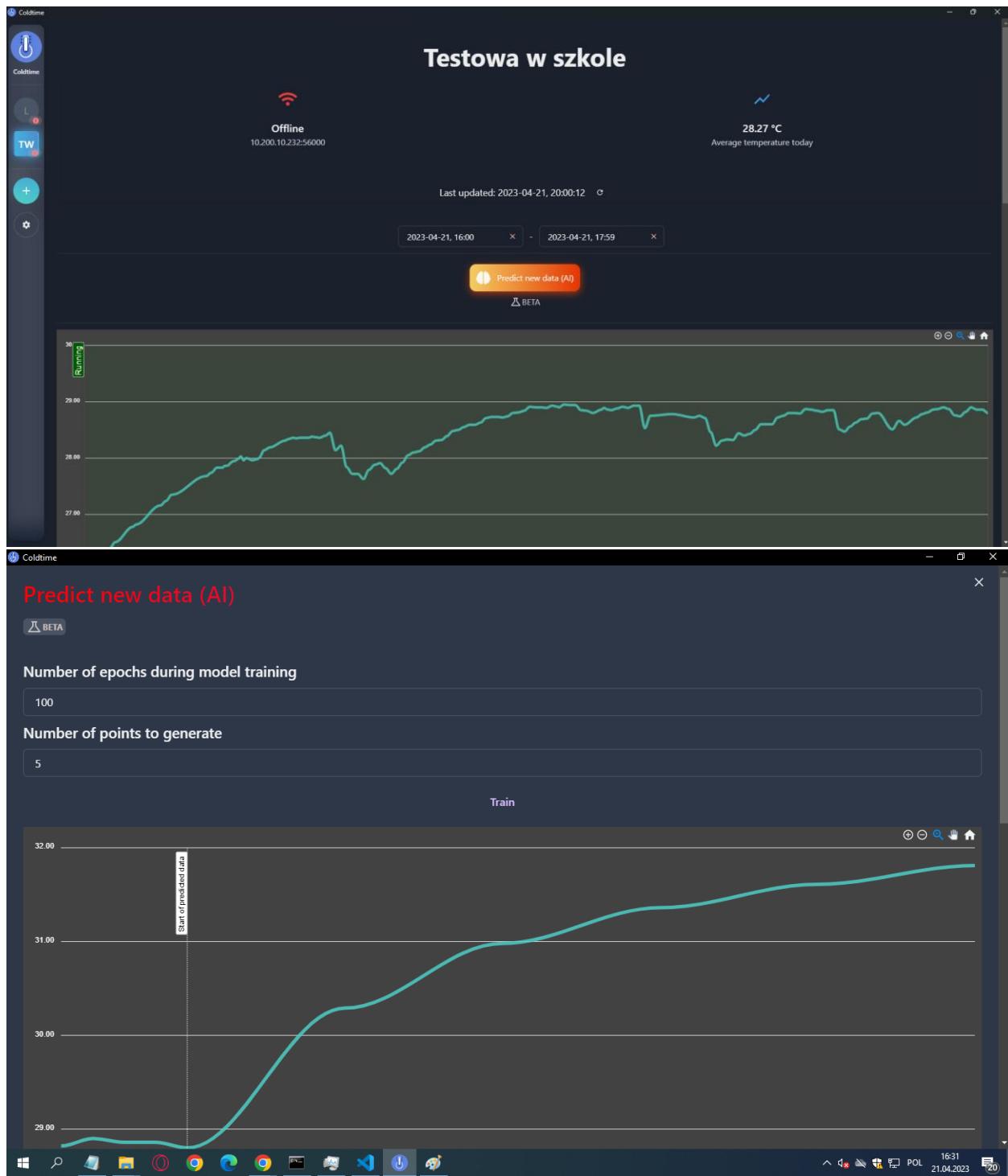
Zrzut ekranu 9 - W przypadku wprowadzenia nieprawidłowych danych aplikacja nas o tym poinformuje



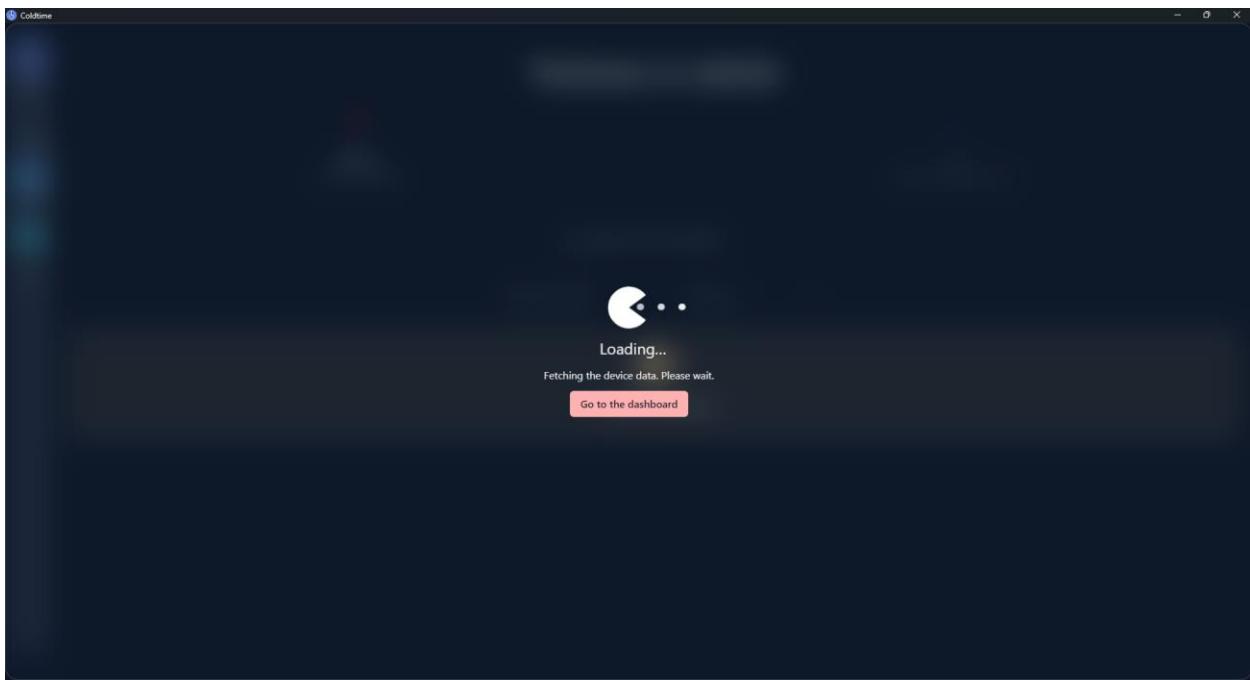
## Obsługa wielu języków

Jak można było zauważyć na poprzednich zrzutach ekranu, były one ukazane w języku polskim. Jednak aplikacja (w całości) jest przetłumaczona również na język angielski, co ilustruje kilka przykładów:





Coldtime – dokumentacja. Autor: [Artur Nowak](#)



A screenshot of the "Settings" screen of the Coldtime application. On the left side, there is a vertical sidebar containing five circular icons: "Coldtime" (blue), "TW" (green), "TU" (grey), "0" (red), and a plus sign (+). Below the sidebar is a gear icon and the word "Settings". Underneath that, the heading "Select language" is displayed, followed by two language options: "Polski" (Polish) and "English". The main content area contains three sections: "Check interval", "Save interval", and "Max data points". Each section has a brief description and two input fields for "Minutes" and "Seconds". For "Check interval", the values are 0 minutes and 5 seconds. For "Save interval", the values are 1 minute and 0 seconds. For "Max data points", there are no visible input fields.



# Jak uruchomić aplikację na bazie kodu źródłowego lokalnie?

Jeśli życzą sobie Państwo, aby aplikację uruchomić w środowisku deweloperskim, tzn. nie w wersji na produkcję (produkcja = standalone, z instalatorem lub plik .exe; development = przez samodzielne “skompilowanie” (w uproszczeniu) projektu), przez kod źródłowy:

Wymagania:

- Node.js (tworzone na 18.13.0 oraz 18.16.0)

Po pobraniu kodu źródłowego, w katalogu z kodem należy uruchomić następujące komendy:



```
# Instalacja bibliotek do developmentu - wymagana
# komenda, aby kolejne działały
npm install

# Utworzenie czystej bazy danych (SQLite) -
# zastosowanie migracji
npx prisma migrate deploy

# Uruchomienie - wersja development
npm run dev

# Utworzenie instalatora na system Windows -
# produkcja
npm run build:win
```



## Dodatkowe pomysły na przyszły rozwój

- Ulepszyć sztuczną inteligencję – patrząc przez pryzmat Państwa firmy, mogłaby ona występować np. w formie dodatkowej, płatnej subskrypcji
- Opcja eksportowania oraz importowania całej bazy danych
- Więcej statystyk (aktualnie tylko średnia temperatura w danym dniu)
- Więcej języków - w pierwszej kolejności niemiecki (ponieważ zakładam, że to kolejna docelowa grupa Państwa odbiorców) oraz hiszpański (ponieważ się go uczę)
- Ogólna optymalizacja, poprawki drobnych błędów, refactoring kodu

**MIĘGO KORZYSTANIA Z COLDTIME!**

