

The Rumble in the Jungle czyli Delphi vs Xcode



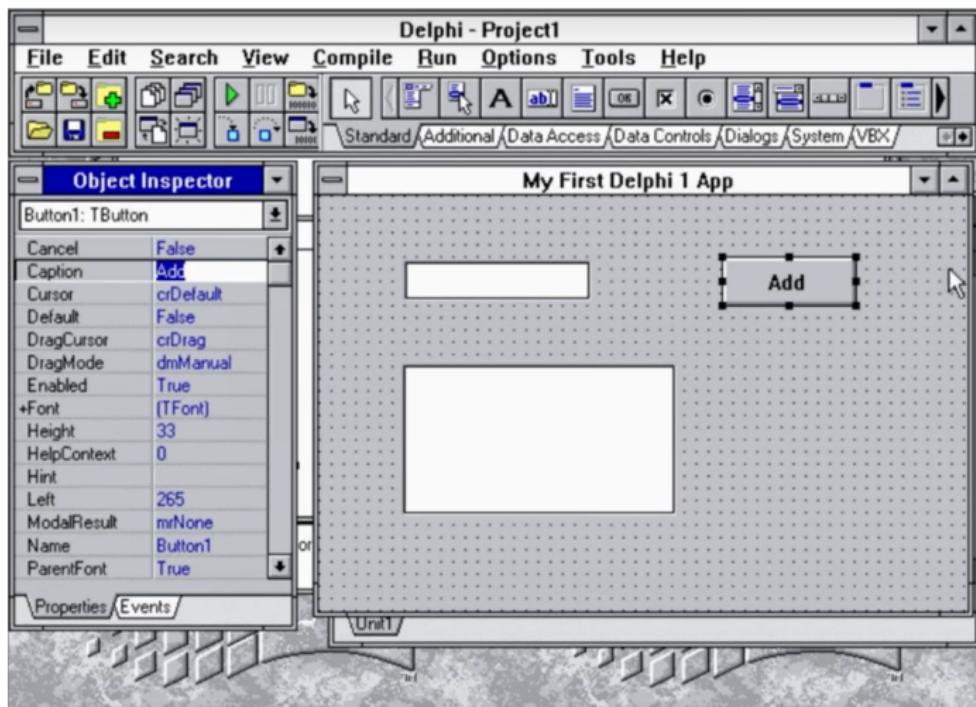
Zdzisław Sroczyński
Caprisoft
Politechnika Śląska



Zlot programistów Delphi 2023

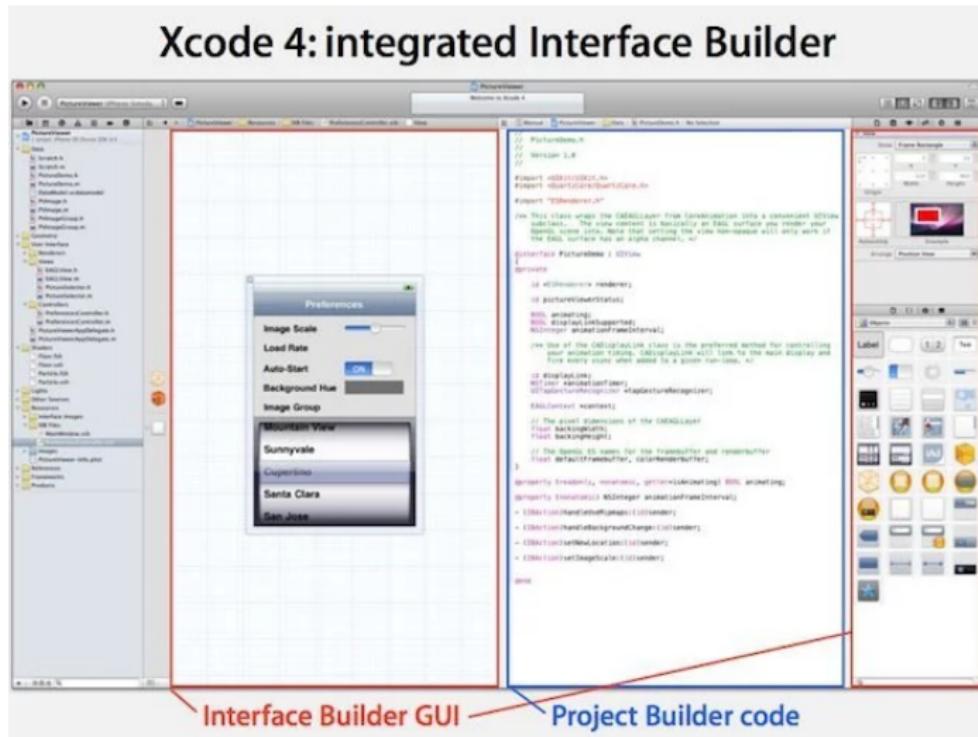
Bohaterowie – Delphi

Delphi 1 1995 (Turbo Pascal for Windows 1991)



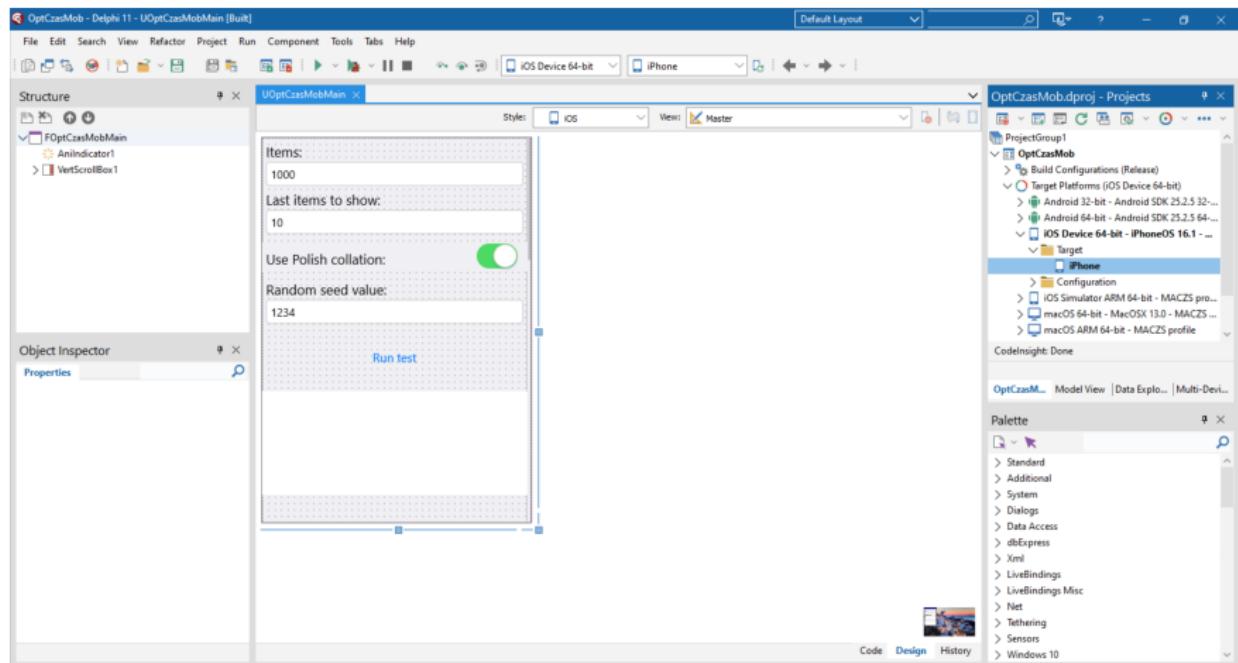
Bohaterowie – Xcode

Xcode 4 2011 (Xcode 1 2003 / tylko macOS)



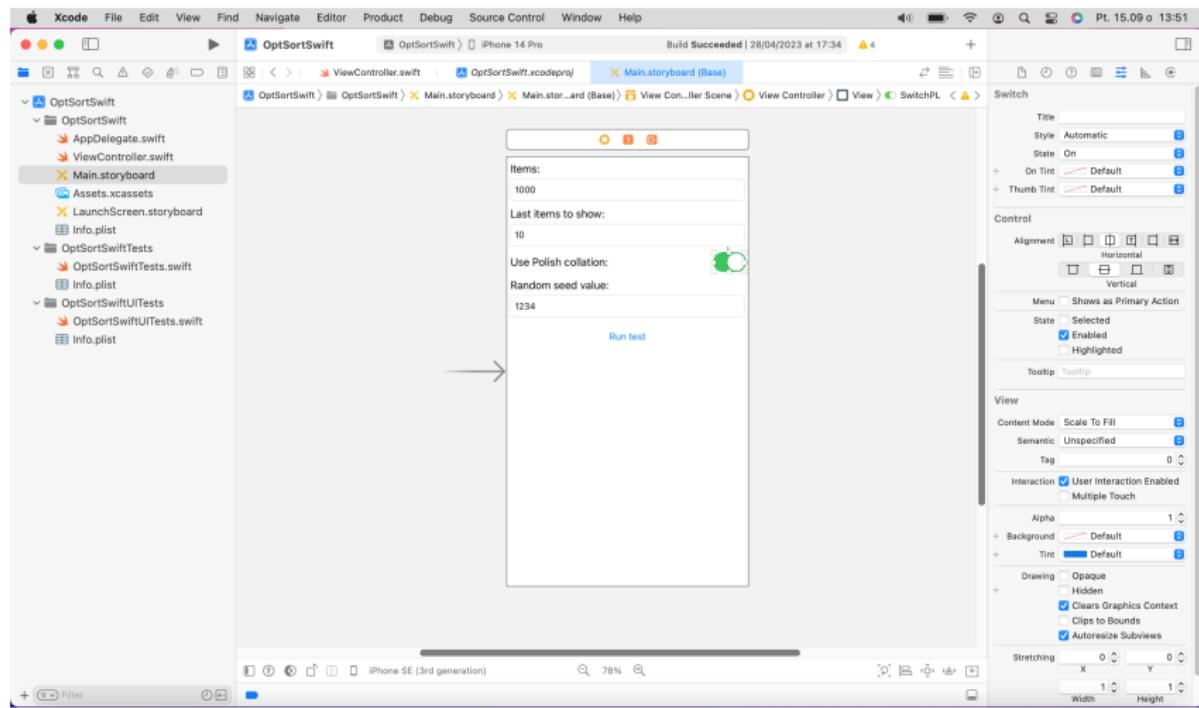
Zamiast wstępu (1)

Prosty projekt, standardowe kontrolki GUI – Delphi

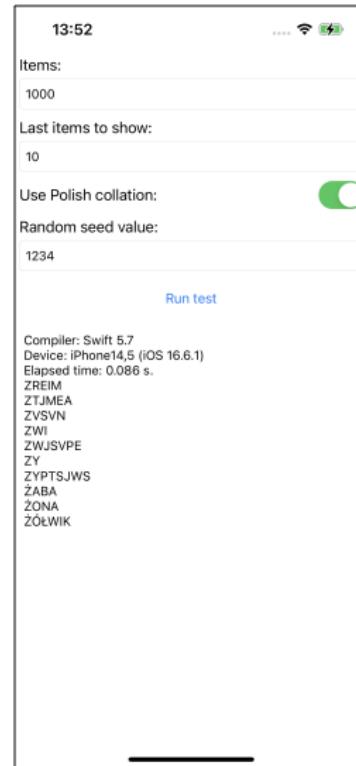
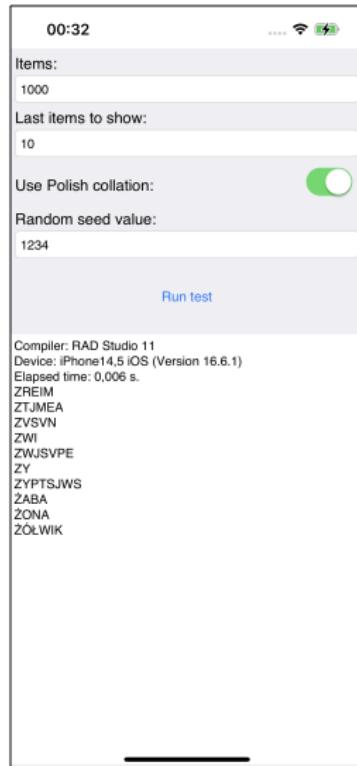


Zamiast wstępu (2)

Prosty projekt, standardowe kontrolki GUI – Xcode



Zamiast wstępu (3)



Delphi – mnóstwo 😊, odrobina 😥 (1)

- ▶ wieloplatformowość (prawdziwa i pełna, włącznie z RTL i filozofią - niekiedy kłopotki: wątki)
- ▶ problemy z symulatorem/debuggerem, ale: łatwe uruchamianie na win32 😊
- ▶ adaptacja nowości w systemie operacyjnym (SO) z opóźnieniem (powszechnie w narzędziach wieloplatformowych)
- ▶ można używać dowolnego API SO, ale wtedy nie jest już wieloplatformowe!
- ▶ ale: niezbyt nadaje się do projektów bardzo mocno zintegrowanych z SO

Delphi – mnóstwo 😊, odrobina 😥 (2)

- ▶ biblioteka FMX jest spójna i w większości wizualna (da się ustawiać właściwości), również jeśli chodzi o komponenty niewidoczne, np. REST
- ▶ wsteczna kompatybilność jest bardzo duża, choć w FMX czasem zdarzają się szaleństwa 😊
- ▶ nie zawsze GUI jest idealnym odwzorowaniem wersji natywnej, choć bywa bardziej czytelne
- ▶ wydajność aplikacji jest bardzo dobra, a wielkość nie jest zatrważająca (sort. 10mln elem. 1,2s Xcode, 6,6s Delphi; z PL: 82s Xcode, 59s Delphi)
- ▶ czasem edytor płyta figle (wszyscy wiemy gdzie i kiedy)



- ▶ obsługa uprawnień nie zawsze konsekwentna
- ▶ FMX nie zna „safe area” na iOS
- ▶ czasem animacje FMX lagują (głównie za 1-szym razem)
- ▶ trzeba uważać na synchronizację i Timer-y
- ▶ obsługa lokalnych baz danych – Firedac, SQLite, Visual Live Bindings (VLB) – wieloplatformowo!
- ▶ komponenty do baz i SQL wzorcowo
- ▶ pułapki: bez dokumentacji dla OnFilledListItem i innych zdarzeń VLB

Xcode – czasem 😊, zdarzają się 😥 (1)

- ▶ biblioteka UIKit i API iOS mają wiele zaszłości z czasów NEXTStep 😊
- ▶ Objective C było przerażające 🦇 Swift mocno się zmieniał z wersji na wersję (już jest lepiej)
- ▶ SwiftUI w wersji rozwojowej wprowadzony jako alternatywa dla UIKit i Storyboard (niewizualny)
- ▶ bez pełnego wsparcia dla nowych funkcjonalności (podpowiadanie, refaktoryzacja!)
- ▶ „wieloplatformowość” polega na obsłudze platform Apple: macOS, iOS, iPadOS, tvOS, watchOS 😊

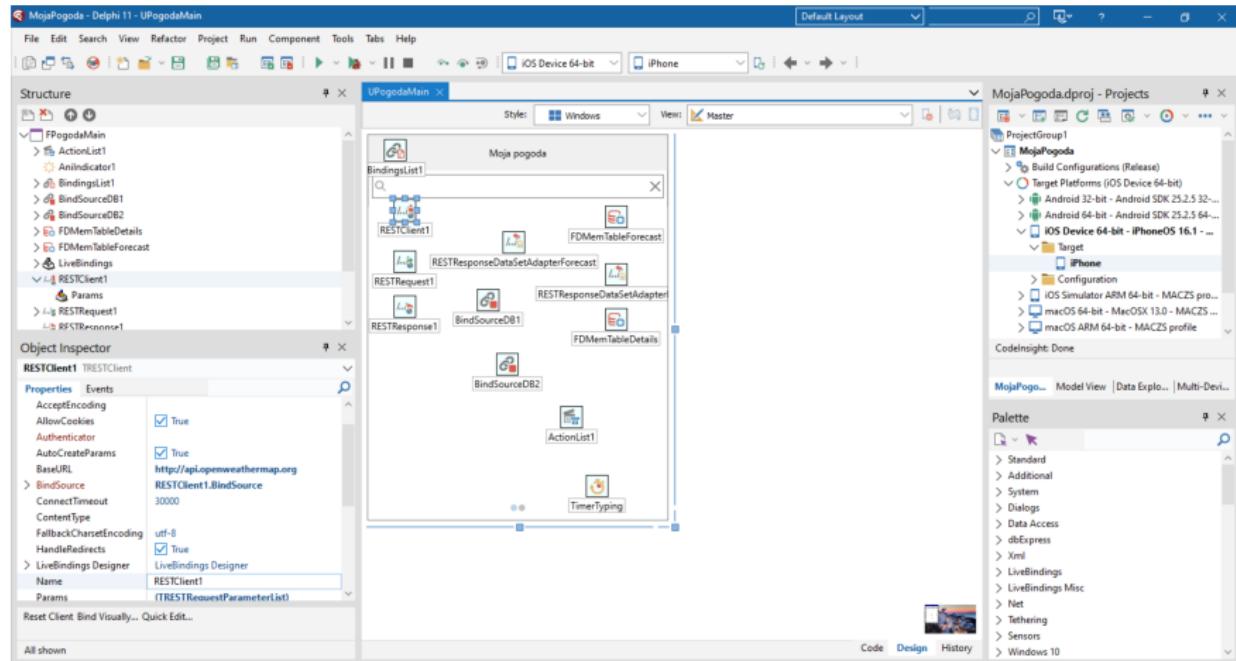
Xcode – czasem 😊, zdarzają się 😥 (2)

- ▶ Interface Builder (IB) - podgląd schematyczny, lepszy niż w B4i, gorszy niż w Delphi, zdarza mu się zaczernić podgląd bez powodu
- ▶ co innego widać w IB, co innego w symulatorze, co innego na urządzeniu (sic!) i to nie są drobiazgi, tylko poważne różnice w implementacji pozycjonowania
- ▶ pożera gigaherce 😊 ale nie lubi, gdy chcemy żeby działało wielowątkowo – potrafi się wtedy zawiesić
- ▶ nigdy nie wiadomo, gdzie co trzeba ustawić: czy w IB, czy w kodzie (delegaty, osadzanie w kontrolerze, dodawanie klasy do widoku!)

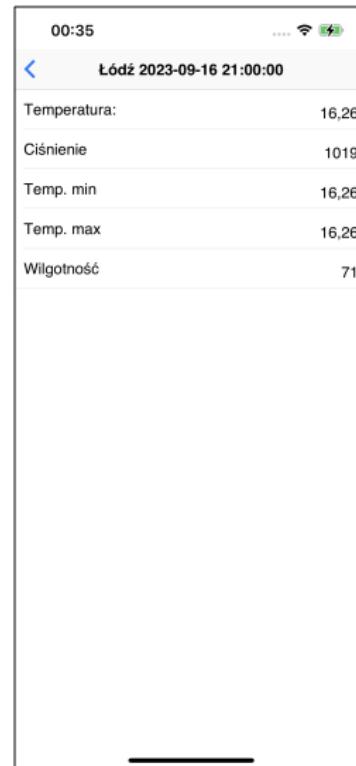
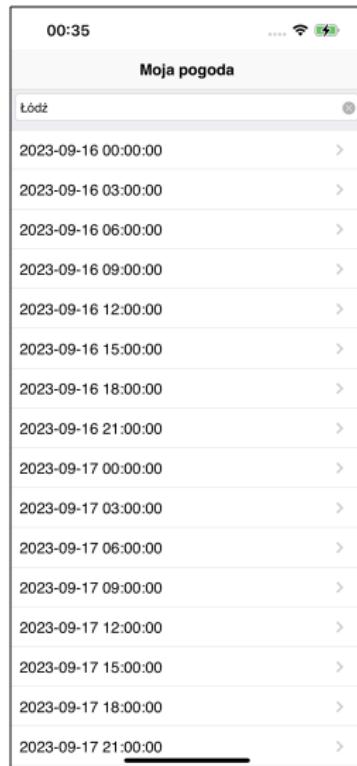
Xcode – czasem 😊, zdarzają się 😥 (3)

- ▶ Core Data – wygodny mapper obiektowo-relacyjny (zapisuje wewnętrznie w SQLite)
- ▶ 90% trzeba zapisać w kodzie, bez szablonu/tutoriala... trudne?
- ▶ obsługa update, delete i refresh w Core Data uciążliwa
- ▶ obsługa filtrowania – ręcznie („ręczny” indeks)
- ▶ wydajność niska, ale przecież nie służy do przetwarzania dużych ilości informacji 😊
- ▶ trzeba uważać na synchronizację/wątki

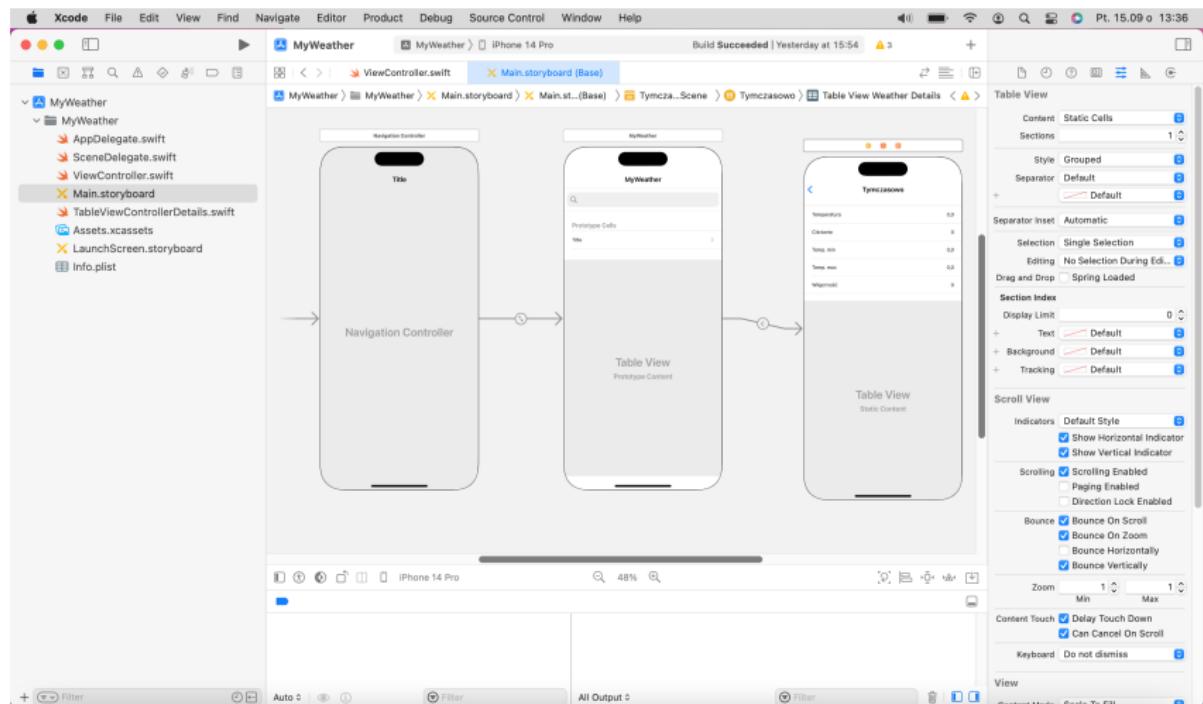
Konsumujemy REST – Delphi



Konsumujemy REST – Delphi na iOS



Konsumujemy REST – Xcode



Konsumujemy REST – XCode na iOS

The image displays two screenshots of an iOS application interface. Both screenshots show the same basic layout with a header, a search bar, and a list of items.

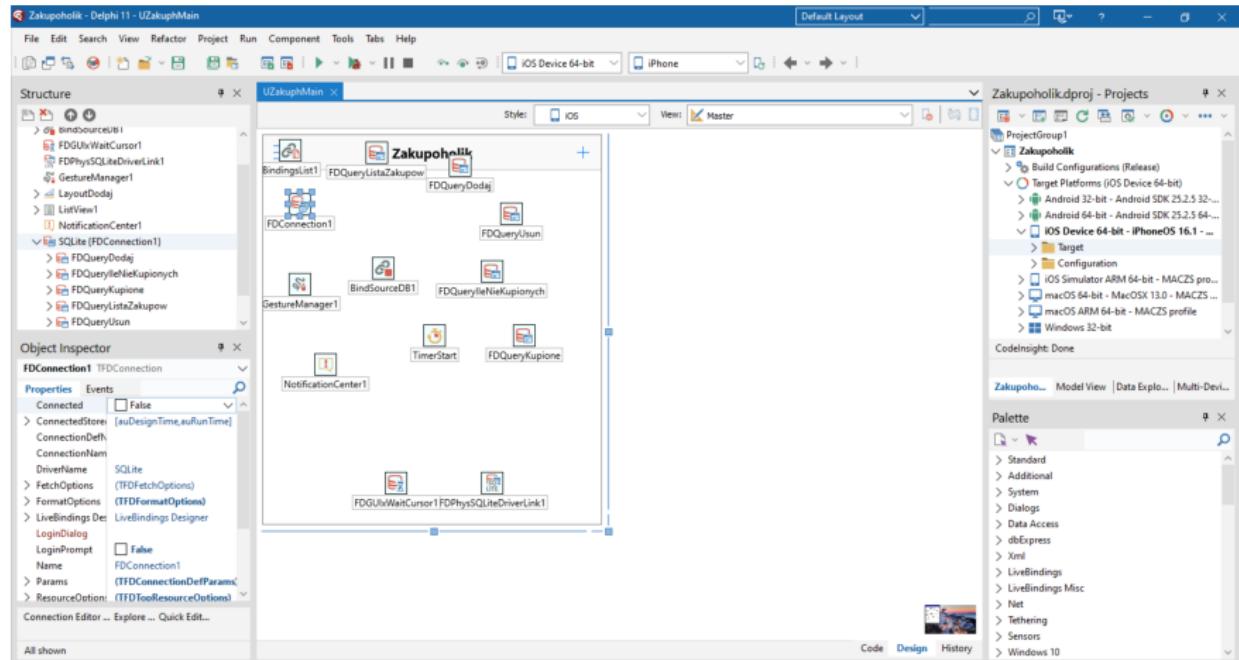
Screenshot 1 (Left): The title is "MyWeather". The search bar contains "Łódź". Below the search bar is a list of timestamps from "2023-09-15 12:00:00" to "2023-09-17 06:00:00", each preceded by a right-pointing arrow.

Timestamp	Action
2023-09-15 12:00:00	>
2023-09-15 15:00:00	>
2023-09-15 18:00:00	>
2023-09-15 21:00:00	>
2023-09-16 00:00:00	>
2023-09-16 03:00:00	>
2023-09-16 06:00:00	>
2023-09-16 09:00:00	>
2023-09-16 12:00:00	>
2023-09-16 15:00:00	>
2023-09-16 18:00:00	>
2023-09-16 21:00:00	>
2023-09-17 00:00:00	>
2023-09-17 03:00:00	>
2023-09-17 06:00:00	>

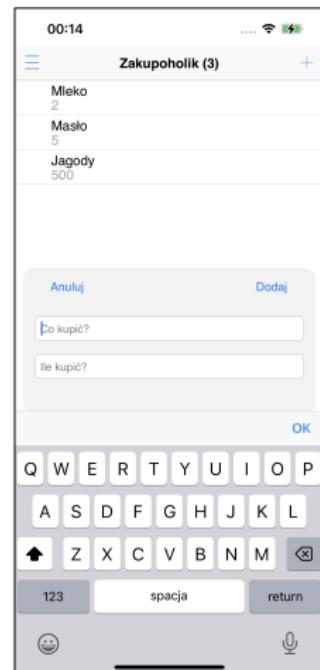
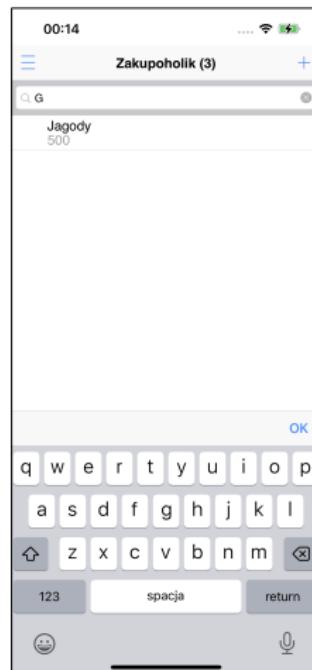
Screenshot 2 (Right): The title is "Łódź 2023-09-15 15:00:00". The list shows the following data:

Parameter	Value
Temperatura	20,13
Ciśnienie	1023
Temp. min	20,13
Temp. max	20,48
Wilgotność	47

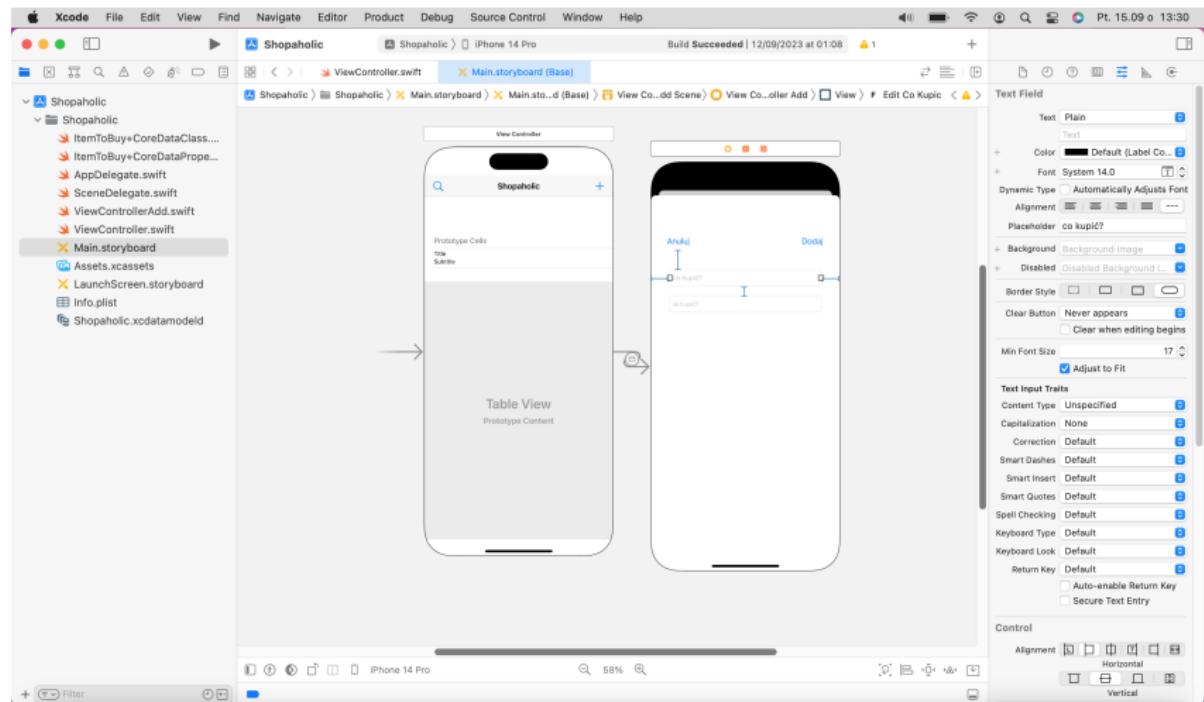
Lokalne dane – Delphi



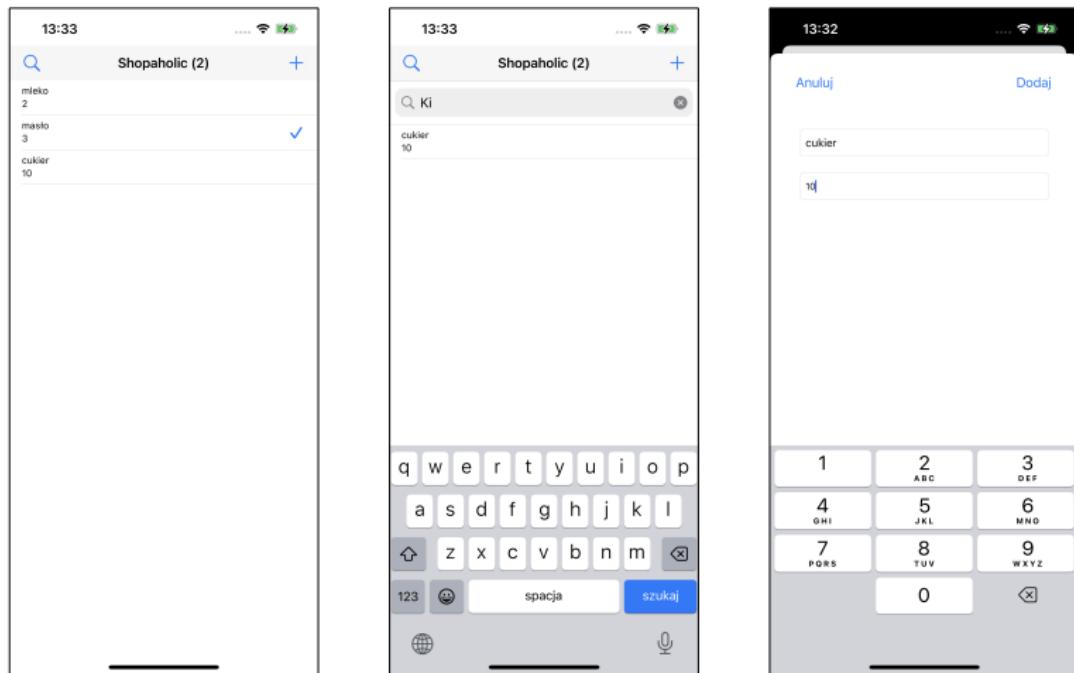
Lokalne dane – Delphi na iOS



Lokalne dane – Xcode



Lokalne dane – Xcode na iOS



Zamiast podsumowania

- ▶ nie będzie werdyktu, nie było nokautu 😊
- ▶ programowanie to dobra zabawa?
- ▶ BASIC (Commodore, Simon's, VBA, Visual/B4x), dBase/Clipper, Turbo Pascal, C, C++
- ▶ Javascript, Lua, Python, Java, PHP, Swift, Kotlin, C#



- ▶ Object Pascal/Delphi

Kody źródłowe przykładowych aplikacji będą tu:
<https://github.com/zdzichs/ZlotDelphi2023>

Dziękuję za uwagę

pytania: zdzislaw@sroczynski.pl