

Transkrypcja

[Podcast Draft Rozmowy S02E04: Ten o Low-Code - Developico](#)

[Ł. F] Dzień dobry, dobry wieczór. Witamy cię w drafcie rozmowy.

[M. G] To podcast o biznesie, o życiu i o IT.

[Ł. Z] Nie zawsze na poważnie, ale zawsze na serio.

[Ł. F] Do wysłuchania kolejnego odcinka zapraszają Cię Łukasz Falaciński,

[M. G] Michał Guzowski,

[Ł. Z] i Łukasz Zajączkowski.

Kim jest deweloper? Czy na rynku brakuje deweloperów?

Deweloper, czyli osoba, która poza programowaniem zajmuje się również tworzeniem nowych koncepcji – rozwiązań. W kontekście low-code i tworzenia aplikacji mówimy raczej o deweloperach. Nie każdy programista musi być deweloperem. Nie każdy informatyk musi być programistą. W 2018 roku, tak samo jak współcześnie, na rynku występowały duże braki deweloperów. W Europie było to pół miliona, a w Polsce około pięćdziesięciu tysięcy. Mimo pojawienia się na rynku nowych deweloperów, popyt na ich usługi również wzrósł dzięki informatyzacji firm wywołanej konkurencją na rynku. Citizen developer to osoba, która nie musi posiadać wykształcenia informatycznego, jej głównym zajęciem jest tworzenie nowych rozwiązań. Jest to możliwe dzięki narzędziom o niskim progu wejścia, niewymagających obszernej wiedzy informatycznych – tak zwany low-code lub no-code. Citizen developer różni się od zwykłego developera tym, że nie musi być programistą.

[Ł. F] Po raz kolejny się mylę, to coś musi oznaczać. Słuchajcie, pomyśleliśmy, że dzisiaj zrobimy mały wyjątek od naszych głębokich przemyśleń i różnych dziwnych dygresji i spróbujemy pogadać o rzeczach bardzo technicznych, a może nie bardzo technicznych, zobaczymy. Tematem dzisiejszego naszego spotkania będzie low-code. No i nie sposób nie zacząć od oczywistego pytania, z czym wam się kojarzy ten low-code i jaki to ma związek po prostu z kodowaniem? Czy to jest tylko i wyłącznie technologia dla deweloperów? A dobra, inaczej, bo widzę wasze zakłopotanie na twarzach.

[M. G] Bo nie wiemy kto ma zacząć, bo jak zaczniemy gadać, to się nie zamkniemy.

[Ł. F] To zacznijmy inaczej: deweloper.

[Ł. Z] Kim jest? Kim jest?

[Ł. F] Dokładnie. Dlaczego potrzebna jest na rynku taka ilość deweloperów i czy mamy ich wystarczająco?

[M. G] Nie mamy ich wystarczająco.

[Ł. F] Znaczący mówiąc „my” mam na myśli rynek, firmę, organizację.

[M. G] A to mamy wystarczająco. Nie, nie mamy, na świecie ich brakuje. Nie wiem dokładnie ile, nie znam liczby na świat. Znam liczbę na Europę, to jest pół miliona i w Polsce 50000, ale to było stan na 2018 rok, więc prawdopodobnie dzisiaj już jest ich znacznie, znacznie więcej, ale znacznie więcej brakuje, natomiast zdecydowanie problem jaki generuje taka ogromna przepaść w niedoborze tych programistów to jest fakt, że praktycznie wszystkie branże się informatyzują w każdej branży potrzeba jakiejś aplikacji, jakiejś automatyzacji, jakiejś konfiguracji sieci czy usług w organizacji, choćby postawienie strony internetowej albo sklepu ecommerce'owego. No to zawsze powoduje, że jakiś informatyk będzie potrzebny i ponieważ każdy biznes już teraz w jakiś sposób musi być...

[Ł. F] Sorry, muszę przerwać, bo użyłeś ciekawego zdania i to moim zdaniem też jest taki mit, który często pokutuje, że potrzebny jest informatyk.

[M. G] Tak.

[Ł. F] Czy każdy deweloper jest informatykiem, czy każdy informatyk musi być deweloperem? Co sprawia, że jesteś informatykiem?

[M. G] To dobre pytanie.

[Ł. Z] No na pewno nie każdy informatyk musi być deweloperem,

[M. G] Może być administratorem.

[Ł. F] No dobra ale co to oznacza, że jesteś informatykiem w takim razie.

[Ł. Z] Zajmujesz się technologią.

[M. G] Tak.

[Ł. F] Technologią?

[Ł. Z] W moim rozumieniu umiesz skonfigurować, obsłużyć plastycznie technologię, bo niekoniecznie umiesz program, który na przykład obsługuje jakiś proces biznesowy, no jeżeli ja umiem CRMa to nie Jestem informatykiem, jeżeli umiem po prostu klikać w aplikacji, która jest CRMem w firmie swojej, no to nie jestem informatykiem.

[M. G] Ale jeżeli konfigurujesz Excela i umiesz pisać funkcję to jest informatykiem?

[Ł. Z] Nie.

[Ł. F] Ale czego ci brakuje, żeby być informatykiem?

[Ł. Z] Dyplomu.

[M. G] Ok i w ten oto sposób mamy jasną definicję kim jest informatyki. Jak nie macie dyplomu, nie jesteście informatykami. Ale w sumie mi się ta definicja podoba, bo to od razu stawia nowe pytanie: czy w takim razie jest taki niedobór informatyków, czy może po prostu konsultantów i informatyków?

[Ł. Z] No właśnie zmierzałem dokładnie do tego, do tego sedna tej tarczy.

[Ł. F] A czy informatyk jest konsultantem? To też może być kwestia semantyki.

[M. G] Tak, może być tak.

[Ł. F] Ale czy zawsze jest?

[M. G] Nie.

[Ł. Z] Bo on generalnie konsultuje.

[M. G] Ale nie z klientami.

[Ł. Z] Taki konsultant helpdesku, jakbym szukał wyrafinowanej nazwy, to by był „konsultant od problemów związanych z sieciami informatycznymi firmy”.

[Ł. F] Nie no wszyscy generalnie, cały biznes jakkolwiek polega na rozwiązywaniu problemów.

[Ł. Z] No dobra, to kim jest informatyk, bo ja nie wiem ciągle. Według was.

[M. G] Inżynierem, który konfiguruje narzędzia.

[Ł. Z] Dyplomowanym inżynierem?

[M.G] Nie, nie musi mieć dyplomu, żeby być informatykiem, natomiast jednak konfigurujesz tą technologię, czyli nie wiem, ustawiasz sieci, konfigurujesz serwer albo programujesz aplikację - jesteś informatykiem.

[Ł. F] Czyli ten wasz przykład z Excel, to co?

[M.G] To jest dobry przykład, bo nie nazwałbyś w teorii informatykiem, chyba ja nie nazwał informatykiem osobę, która umie robić Excel, nie? Tak jak nawet już mówiąc, używając pewnej frazy, której jeszcze nie wyjaśniliśmy, czyli citizen developer to nie jest informatyk.

[Ł. Z] No nie jest, właśnie.

[M.G] A jednak buduje profesjonalne rozwiązania.

[Ł. Z] Więc czy możemy przejść w takim razie, że deweloper to jest ta osoba, która rozwiązuje problemy? Używa do tego narzędzi IT, tak jak reszta w dużej mierze biznesu, ale jednak ten jakby jest stopień zaawansowania tych narzędzi i wiedza potrzebna do tego, żeby tych narzędzi użyć oraz, to co tworzymy, to to jest jakby to rozróżnienie. Teraz wracając do przykładu z Excel. No jak jestem kumaty i siedzę w tej księgowości i znam tego Excela i widzę, że w mojej firmie przydałoby się zautomatyzować jakiś proces - nawiązując do poprzedniego naszego odcinka - i potrafię to zrobić w Excelu, to kim ja jestem w zasadzie, to jestem księgowym?

[M.G] Ja bym powiedział, że jeżeli powiedzmy pracujesz w dziale księgowości i masz tego Excela wykorzystywanego w jakiś tam procesie w księgowości, no to jesteś księgowym, ale jesteś także citizen developerem.

[Ł. Z] Kim bym był?

[M.G] Kim jest citizen developer? Ja bym powiedział taki trochę samouk, który jest w stanie tak skonfigurować program, żeby wykonywał niestandardowe operacje, czyli jest to takie powiedzmy laik informatyczny, który jednak nie boi się technologii i ma w sobie tyle takiej ciekawości, samozaparcia i logicznego myślenia, aby zbudować nową funkcjonalność w programie.

[Ł. F] No i to jest chyba najważniejsze, że są ludzie, którzy mają po prostu taką smykałkę. Jedni to robią majsterkując, inni pisząc makro w Excelu, a low-code to jest taka odpowiedź właśnie dla tych osób, żeby wykorzystać ten potencjał w bardzo konkretnych biznesowych scenariuszach, wykorzystując narzędzia, które nie są zbyt skomplikowane. Ten próg wejścia, to jest chyba kolejna cecha, nie jest jakiś za duży, czy jest?

[M.G] Moim zdaniem jest tak niski, jak jeszcze nigdy nie był w przeszłości.

[Ł. F] No właśnie o przeszłość. Zając, ty pamiętasz jakies low-code?

[Ł. Z] Przeszłość, tak? Wyciągają mnie z formaliny i „Jakie jest pytanie o przeszłość? Z chęcią odpowiem.”

[Ł. F] Dokładnie. Gdy zaczynaliśmy z SharePoint, bo low-code jest teraz takim base wordem, który jest teraz szeroko powtarzany, ale to nie jest nowy pomysł.

[M.G] No na najprostszy jakiś przykład jakies CMSy, czy jakiś WordPress to już jest jakiś no-code, tak naprawdę.

[Ł. F] Ja pamiętam gdzieś tam na początku, pamiętasz InfoPath na przykład?

[M.G] Pamiętam, robiłem w tym aplikacje.

[Ł. F] Sam koncept nie jest nowy.

[M.G] Nie powiem dla jakiej firmy, ale mogę powiedzieć, że obsługuje energię i gaz.

[Ł. Z] No to Power Apps, to to co ostatnio rozmawialiśmy, czyli nałożenie innego koloru i nazwy na dawniejszego CRM.

[Ł. F] O, to mówisz o bardzo konkretnym, dokładnie.

[Ł. Z] Tak, tak dokładnie. InfoPath też był dość konkretny. No wydaje mi się, że to najlepsze jakie było...

[M.G] Sorry, muszę się wciąć. Swoją drogą, jak słucha nas ktoś, kto jeszcze pracuje na InfoPath, ja tak chętnie bym poprosił, żeby pokazać mi jak jeszcze wyglądają te stare infopathy. Mi tak brakuje tych narzędzi, żeby na nie popatrzeć, bo one miały taką prostą, wiecie, taką strukturę cepa. Mi się tak strasznie podobały. Ja darzę je sentymentem, więc jeżeli słucha nas ktoś, kto pracuje na InfoPatch niech mi da znać. Ja chętnie popatrzę. Pocieszę się do tego.

[Ł. Z] Jest najlepiej, jak jest do tej pory, no bo to wszystko idzie do przodu i myślę, że będzie jeszcze prostsze. Może prostsze niż sobie wyobrażamy dzisiaj, a tak de facto, no to jest to jakiś nowy sposób opakowania pomysłów, które już były i były trochę inaczej wykorzystywane. Ja sobie tak to wyobrażam. Taki infopatch nawet wspomniany, to już był wtedy szal, że my wszystko - wyklikamy i już nie trzeba będzie. Ten formularz już ma maski i filtrowania, może tyle za nas sprawić, zautomatyzować.

[Ł. F] I Ogarniemy.

[Ł. Z] I w pewnym momencie: Ach, no a tego nie może. No to trzeba pisać i też dochodzimy do tego, że te wszystkie - chyba mieliśmy już taką rozmowę - czy to jest low-code czy no-code?

[M.G] Ale poczekaj, bo wspomniałaś o InfoPath, że on już był, to dlaczego się nie utrzymał, skoro był low-code w sumie?

[Ł. Z] No to może znasz Fala historii? Nie wiem.

Problemy low-code w przeszłości.

W przeszłości dostawcy aplikacji low-code korzystali z różnych narzędzi, co często uniemożliwiało implementację dwóch rozwiązań od różnych deweloperów. Dzięki standaryzacji, współcześnie technologia low-code jest popularna wśród zlecniodawców. Przewiduje się, że zapotrzebowanie na aplikacje low-code i no-code znacznie wzrośnie w najbliższych latach. Dotyczyć to może usług związanych z tworzeniem stron internetowych, narzędzi do automatyzacji, do obsługi płatności czy narzędzi wewnętrznych i wielu innych.

[Ł. F] To nie będzie historia InfoPatch, to moim zdaniem to, co się zmieniło, to to, że mamy coraz więcej standardów, że coraz więcej providerów dużych usług korzysta z podobnego sposobu implementowania swojego wszechświata i to co kiedyś było niemożliwe, czyli nie wiem, zintegrowanie rozwiązania zrobionego w Microsoft w prosty sposób taki low-code z rozwiązaniem zrobionym przez innego providera, to tego po prostu zrobić się nie dało. To była ściana. Po prostu te ściany się przestawiły troszeczkę dalej. Ale one cały czas też są. Nawiązując do tego, co mówił Zając, że to oczywiście te rozwiązania low-code są super, ale częsty błąd jaki jest tutaj popełniany - tak mam wrażenie - że bardzo entuzjastycznie do nich podchodzimy, nie sprawdzając, gdzie są ich limity. No bo się na oglądamy, że w zasadzie w takim rozwiązaniu XYZ to mogę sobie zrobić ERPa, a tutaj CRMa, no a tutaj obieg faktur. To jasne, jest możliwe pod wieloma, wieloma założeniami, ale odpowiadając na to poprzednie pytanie jest standardem moim zdaniem.

[M.G] Czyli wcześniej nie było takiej standaryzacji tej warstwy komunikacyjnej jaka jest teraz. Ja się pod tym absolutnie podpisuję.

[Ł. F] To pewnie jest konsekwencja z kolei pojawienia się chmury bardziej, niż zmian w samym samym low-code.

[M.G] No, ale myślę też, że jednak bardzo dużo zrobiła otwartość, otwarty umysł i zmiany też na poziomie strategicznym takich organizacji jak Microsoft, który no kiedyś to tak powiedziałbym, że był trochę taką ślepą determinacją. Próbował forsować swoje standardy, swoje podejście do jakiś tam różnych standardów, a po jakimś czasie zrozumieli, że nie ma co się kopać z koniem zaczęli implementować cudze rozwiązania i gitara i nie było z tego powodu tragedii, a nawet zyskali na tym.

[Ł. F] A jakieś przykłady, bo mówimy, myślę, że dla kogoś, kto nie siedzi w Microsoftzie w tej technologii, to może się wydawać, że rzucamy jakimiś dziwnymi nazwami. Nie wiadomo czego, a takie bardziej przyziemne, przyziemne platformy low-code, które można by tak zaklasyfikować, które jest szansa, że zna szersze grono. Może jakieś pomysły?

[M.G] No my oczywiście siłą rzeczy w platformach Microsoft siedzimy, więc Microsoft Power Apps.

[Ł. F] A WordPress na przykład nie wydaje ci się, że można by go podciągnąć też?

[M.G] Absolutnie tak. Znaczący, tych platform jest w ogóle multum na dobrą sprawę. Jeżeli byśmy chcieli sobie podyskutować o tym, jakie są platformy low-code czy no-code, to moglibyśmy śmiało w ogóle zacząć od tego, że rynek low-code jako taki, to jego wartość diametralnie się zwiększa i jest to oczywiście odpowiedź na zapotrzebowanie, ale sam rynek był wyceniany na 46 miliardów dolarów w 2026 roku i podwoi się w ciągu kolejnych 2 lat, co pokazuje właśnie ogromne ssanie. Mało tego - Gartner, duża taka firma audytująca rynek przedstawiła w jednym ze swoich raportów, że do 2025 roku 70% wszystkich nowych powstających aplikacji dla biznesu będzie oparta o low-code albo no-code. I tutaj, jeżeli znowu mówimy o low-code, to musimy sobie zdać sprawę, że tych kategorii jest bardzo dużo, bo mamy zarówno jakby takie web buildery, czyli do budowania witryn „www”, mamy buildery do aplikacji, mamy jakieś narzędzia wewnętrzne, narzędzia do automatyzacji, do obsługi płatności. I mamy narzędzia do analityki, mamy narzędzia do budowania formularzy.

[Ł. Z] RPA.

[M.G] Tak, RPA - Robotic Process Automation - więc tych kategorii jest mnóstwo i w każdej z tych kategorii mamy bardzo dużo, wiele rozwiązań, zarówno w takich dużych organizacjach, typu Microsoft czy Google, ale również niezależnych. Na przykład, jeżeli mówimy o automatyzacji procesów, no to Microsoft ma Power Automate, Google tam z tego co wiem to rozpoczyna rozwój swojego AppSheet, żeby on poza aplikacjami robił też procesy, a na przykład niezależnymi popularnymi narzędziami jest Integromat, czy Zapier, tak? Albo jest IFTTTA. Web buildery, to najpopularniejszy WordPress. A jeżeli byśmy chcieli sobie wejść znowu w świat Microsoft, no to mamy tutaj Swaya albo SharePointa.

Dla kogo jest low-code?

Osoba zajmująca się citizen developmentem to ktoś, kto dobrze zna procesy, które chce usprawnić. Powinien być wyspecjalizowany w tym, co chce usprawniać. Poza znajomością procesów, musi poznać narzędzia, dzięki którym będzie to możliwe. Na szczęście próg wejścia w low-code jest niski, a jego wybór może okazać się korzystniejszy, niż nauka programingu od podstaw.

[Ł. F] No właśnie, więc jest duża szansa, że nawet jeżeli wcześniej nie spotkałeś się nasz drogi słuchaczu z tym terminem, to jest duża szansa, że korzystasz z tych narzędzi, bo one są tak powszechne i wszędzie wokół nas. I patrząc na to, jak szybko rosną zapotrzebowania, nie tylko biznesu, jak szybko trzeba te nowe funkcje, aplikacje wdrażać. No to jest nie do uniknięcia. To co powiedziałaś, te 70% rozwiązań, które gdzieś tam ten low-code wykorzystują, bo to też moim zdaniem tak na marginesie ważna rzecz, że to nie jest tak, że musimy wszystko robić w low-code i koniec, tak? I się na to zamykać. Jak najbardziej gdzieś tam te rozwiązania pisane mogą być fajnym uzupełnieniem. To jak najbardziej tak może być. A zmieniając kontekst, tak patrzę na Ciebie, dla kogo jest ten low-code? Kto byłby takim idealnym kandydatem, a kto by się dobrze odnalazł takich takich technologiach jak citizen developer?

[Ł. Z] Heavy user, czyli ktoś, kto zna dobrze środowisko, w którym siedzi i zaczyna mieć zrozumienie głębszych potrzeb. Wyobrażam sobie dowolną osobę w biznesie, która korzysta z jakiejś aplikacji mocno, na przykład siedzę mocno w CRMie, siedzę mocno

w fakturach, siedzę mocno w Excelu, znam proces i narzędzia, czyli zaczynamy rozumieć.

[M.G] Proces i narzędzia? Czyli dwie rzeczy trzeba umieć?

[Ł. Z] Zaczynam moim zdaniem, Ja sobie tak wyobrażam idealną osobę, bo znajomość narzędzia też pokazuje, że ja chcę i umiem ogarnąć narzędzie. No umówmy się jak ktoś umie dobrze Excela to nieważne czy jest informatykiem, deweloperem czy nie ma nazwy, to znaczy, że naprawdę ogarnia. Excel jest olbrzymi, jak znamy jakikolwiek system dobrze, to pewnie jak umiem się nauczyć, jak to od strony informatycznej - tu nie mówimy o trzewiach, bo to warstwa low-code jak dla mnie, to nie trzeba rozumieć wszystkiego pod spodem, wystarczy rozumieć moduł – puzzles, z których to się składa no to taka osoba plus znajomość biznesu. Czyli osoba najbliższej i najgłębiej w procesie i w technologii, która ten proces obsługuje. Tak sobie to wyobrażam.

[Ł. F] A my to zauważamy z pierwszej ręki, że coraz więcej osób szuka swojego miejsca na rynku pracy. Siedzą w jakiś takich typowych biznesowych działach i z jednej strony szkoły, bootcampy, jakieś deweloperskie, a ludzie skuszeni perspektywami niesamowitych zarobków. No to się dzieje wokół nas. To jak to ma się do low-code'u? Czy to jest jakaś alternatywa? Uzupełnienie? Realna ścieżka rozwoju, kariery?

[M.G] Dobrze to nazwałaś - realna ścieżka rozwoju, bo ja niestety często widzę przypadki, gdzie na przykład człowiek, który przez całe życie siedział w jakiejś branży, nie wiem, finansowej czy jakimś marketingu nagle wchodzi w pythona robiąc sobie jakiś kurs pythona, bo python jest teraz najbardziej sexy. Więc uczy się tego i myślisz, że po roku będzie normalnie devem takim regular devem, który będzie zarabiał 15 K na rękę? Nie i to jest.

[Ł. Z] Albo java.

[M. G] Albo java, tak. Ale python jest teraz absolutnie na topie, bo możesz nim i aplikacji, i analitykę, i tam Wszystko, o Jezu. A prawda jest taka, że no jak masz już wiele doświadczenie w jakiejś branży, to takie zupełne wychodzenie z niej, to jest trochę nie szanowanie swojego doświadczenia i czasu, jaki spędziłeś w tej branży i jaki poświęciłeś na poznanie procesu, więc dużo lepiej byłoby, gdybyś jednak skupił się na przykład na technologii, która jest prostsza w rozpoznaniu, ale z kolei bardziej wymaga tej znajomości merytorycznej, tej branżowej, czyli na przykład jak jesteś pythonowcem, no to zanim zaczniesz robić te swoje aplikacje dla marketingu, powiedzmy coś po marketingu, no to minie tych parę lat. Ale jeżeli jesteś po marketingu i dodatkowo byś wszedł w technologię low-code, to nie wiem, do automatyzacji procesów marketingu czy budowanie aplikacji dla marketingu to jesteś w stanie w 6 miesięcy stać się normalnie już takim regular developerem czy citizen developerem, który jest w stanie te rozwiązania budować niejednokrotnie znacznie sprawniej niż taki zwykły deweloper.

[Ł. F] Czyli reasumując, to nie tylko rozwiązania powstają szybciej, ale też łatwiej zaadoptować te narzędzia po prostu od strony i twórcy, a jak ktoś jest - no bo po coś te rozwiązania robimy, nieważne czy to jest jakieś rozwiązanie w pythonie wyrzeźbione, czy jakiejś innej technologii czy low-codzie, to jednak robimy to dla biznesu, dla kogoś. No oczywiście biznes no to jest uogólnieniem - dla odbiorców. Z ich perspektywy to

ma znaczenie, czy czymś się różnią? Te rozwiązania low-code od tych takich customowych?

[Ł. Z] No to jak jeździ ci się dobrze samochodem, to nie pytasz, jak on był zrobiony.

[M. G] Dokładnie. Znaczą one się niczym nie różnią - aplikacja wygląda jak aplikacja. A czy ona jest była budowana z low-code? No tak jak wiesz to nawet strona wordpressowa.

[Ł. Z] Brzmi to super pięknie. No bo z jednej strony mamy technologię, która pozwala wszystko zrobić szybciej. Łatwiej jest wyszkolić, wyprodukować sobie ludzi albo nawet znaleźć, tak? Bo to też jest ciekawa ścieżka. Można po prostu takich ludzi poszukać w swojej organizacji, którzy coś takiego mogliby zrobić. No to, gdzie jest haczyk?

Ograniczenia low-code.

Low-code, jako metoda korzystająca z gotowych rozwiązań programistycznych, daje mniej możliwości względem pełnego programowania. Często jednak te rozwiązania są wystarczające. Problemem może okazać się opłacenie licencji narzędzi low-code, w szczególności, jeśli nie korzystamy z nich na co dzień.

[M. G] Haczyk jest na pewno w pewnych ograniczeniach tych platform, no bo ponieważ to jest low-code, no to jest troszeczkę jak z klockami Lego, gdzie klocki Lego mają pewnie już swój zdefiniowany kształt. I owszem, z nich możemy robić najróżniejsze, najbardziej skomplikowane budowle. No, ale jeżeli chcemy zejść jeszcze poziom niżej, to nie możemy. Nie wiem podzielić klocka na pół. No on jest niepodzielny, to jest atomowa część. Programując tak w standardowy sposób, jesteśmy w stanie każdy klocek dowolny zbudować, wymyśleć. W low-code takiej możliwości nie mamy, no są pewne ograniczenia plus często jest to związane z jakimiś opłatami licencyjnymi. Wiele platform posiada opcje darmowe, ale tam znowu ograniczone tym, że nie wiem, na przykład na flowach to macie maksymalnie 6 minut na odpalenie, powiedzmy. Różnie no.

[Ł. F] No właśnie, a kasa? Bo to też jest, no pewnie na początku w ogóle wyboru rozwiązania czy ścieżki, w którą idziemy, żeby dowieść jakąś funkcję. No to kluczowa sprawa: wiele osób sądzi, że - według mnie to to nie jest takie proste i prawdziwe - low-code, skoro jest prostszy i szybszy, to musi być tańsze. Że to zawsze jest tańsze? Nie. Wiemy, że nie jest tańsze, to kiedy jest droższe?

[M. G] Tutaj wydaje mi się, że low-code staje się droższy w szczególności wtedy, kiedy zaczyna być, albo bardzo, bardzo duży i finalnie rzeczywiście opłacałoby się to bardziej przenieść na niezależną platformę. W sensie normalnie, to zbudować jakoś tam pythonem właśnie albo wtedy, kiedy korzystamy z niego zbyt mało volumenowo, czyli na przykład płacimy za licencje dające nam 100 flow do wykorzystania a wykorzystujemy 2.

[Ł. F] A płacimy za 100.

[Ł. F] No właśnie, bo to chyba wszystko się sprowadza do tego, że to bywa droższe jak nie dopasujemy tego rozwiązania do modelu licencyjnego, bo to to jest coś, co w każdej jakimś rozwiązaniem low-code występuje, że jest pewna zawiła matematyka - bo wiem, że Zajac się zajmuje licencjami, więc wie jak to jest skomplikowane - którą

trzeba rozwikłać w przypadku takiego budowanego rozwiązania bywa łatwiej, bo mnożymy MDeje, czy tam godziny tego dewelopera i dostajemy coś, jeżeli robimy to oczywiście dla siebie.

[Ł. Z] No chyba i to, i to podejście ma perspektywy błędów. No bo nawet jak piszemy apkę, to możemy po pierwsze nie doszacować, po drugie może się przeciągnąć projekt, po 3 nie weźmiemy pod uwagę - co jest dość istotne - obsługi administracyjnej, bo potem rzadko umiemy to sami obsługiwać jak ktoś nam napisał, koszty serwerów. I tak można pomylić się faktycznie, no prawda jest taka jak źle te koszty policzymy, to nam wyjdzie zły wynik. Będziemy zaskoczeni, że to dużo kosztuje, ale przy dobrym liczeniu kosztów licencyjnych, to trudno mi sobie wyobrazić, gdzie to jest droższe. Przez to, że to jednak z założenia - mówię o low-code'zie versus kodowanie od zera - kodowanie od zera, jak ktoś mówi, że wszystko można, to właściwie ma rację. W low-code od razu możemy powiedzieć, że będą pewne ograniczenia gdzieś daleko, ale będą i tyle.

[M. G] Też trudno powiedzieć jakie, bo to bardzo zależy. To zależy od profilu. Tak to.

[Ł. Z] Ale gdzieś tam wiemy, że ktoś tam w stanie będzie wymyśleć. Przy obu podejściach dobrze obliczonych kosztowo, to low-code musi być trochę tańszy, bo nie dotknie wszystkich, tylko pytanie, czy tu nawet już nie mówię o zasadzie Pareto, czy faktycznie potrzebujemy 100% wszystkiego? A życie pokazuje, że nie. Nawet my z Falą mieliśmy taką rozmowę, jak na początku wszystko kodowaliśmy, gdzie z kodowania zrobić low-code? Chodzi o mindset, gdzie powiedzieć, że już nie da się, chociaż by się dało, tylko po co? Lepiej zrobić tego mniej w „module low-code” w cudzysłowie, żeby ograniczyć te ciągłe potrzeby, które w pewnym momencie są koncertem życzeń. Bo jak się spyta kogoś czy on to chce, to on ci odpowie nie wiedząc o co chodzi, że on to chce, bo lepiej mieć więcej. Jakby cię spytali, czy pana samochód miałby mieć to, to już nieważne co jest po tym co ja powiedziałem, to ty już to chcesz. Jak to jest w cenie, no pewnie, że ja to chcę. I wystarczy popatrzeć na rosyjskie przeróbki samochodów tuningowane w internecie - spoiler większy od samochodu.

[M. G] Najwyżej nie będę używał.

[Ł. Z] No ale to więc ciekawe pytanie: kiedy jest droższe? Nie umiem sobie tego wyobrazić, ale może po prostu mam za mało doświadczenia.

[M. G] Ja bym na pewno powiedział jeszcze jedną rzecz, którą trochę wspominałem wcześniej, ale ugryzę temat od innej strony. To znaczy, warto też pamiętać o tym, kiedy nawet jak podnosiłem pewne koszty związane z low-code, to kiedy te koszty zaczynają maleć? Czyli na przykład zaczynają maleć wraz ze skalą - im, więc zaczynamy używać tych rozwiązań, na przykład budujemy nie 2 aplikacje, a 20 aplikacji, a to wcale nie jest dużo. No jak wiemy, są instalacje i po 100 aplikacji low-code w organizacji, to te koszty się bardzo rozkładają.

[Ł. F] Mhm, ale to też jest ważna rzecz, właśnie te licencje. Te koszty zawsze będą liniowo związane z ilością licencji. To niekoniecznie musi oznaczać ilość użytkowników. Więc tutaj ta korelacja jest zawsze stała w przypadku tych customowych rozwiązań. Na ogół oczywiście są zawsze jakieś stałe koszty związane z utrzymaniem, ale to czy obecne technologie czy to będzie 100 użytkowników czy 100 tysięcy, no to jest też czasami przesunięcie jakiegoś tam suwaczka w AWSie. Więc wydaje mi się, że low-code, tutaj często błąd polega na tym, że jest właśnie traktowany jako taki prosty, także na tym etapie koncepcyjnym i planowania. No bo chyba się zgodzicie, że co do

zasady ten proces wytworzenia tego rozwiązania w obu podejściach jest bardzo podobny.

[Ł. Z] Ale to też jeszcze trzymając się tych licencji my mamy często w praktyce często sytuację, kiedy klient już za to zapłacił.

[Ł. F] A to jest bardzo ciekawy przypadek. Tak, masz rację, słusznie. Mówimy o sytuacji, w której na przykład posiadamy już ten low-code, a my go nawet nie wiemy, że posiadamy go.

[Ł. Z] Tak. „Droży państwo, licencje macie płacić jeszcze do roku, można zrobić to, to i to.”

[Ł. F] Podpowiem konkretnie, że chodzi o Office'a 365, gdzie Power Apps, Power Automate - te rozwiązania już są, więc jeżeli posiadacie takie subskrypcje...

[M. G] Dowolna licencja Office'a 365 posiada pełen zestaw narzędzi low-code do budowania stron do automatyzacji, do dodawania aplikacji. Te rozwiązania macie w licencji Office'a 365, macie Teams, macie SharePointa Online, macie Power Automate'a, macie Power Apps, macie Power BI, macie Power Virtual Agents.

[Ł. F] No cóż, powiedz to za rozwiązanie?

[Ł. F] Mhm no właśnie, no dobra, to dla kogoś kto nas słucha i czy nie ma tego Office'a, to już tam palicho, to jaki byłby następny krok, co powinien zrobić, jak zainteresował go ten temat?

[Ł. Z] A gdzie jest, w którym miejscu? Po prostu w zerowym, ale się zainteresował? Przede wszystkim, zanim stwierdzi, że warto wdrożyć w firmie, to się doedukować z tego zakresu, poznać lepiej to narzędzie, te narzędzie czy te narzędzia, a jak się uczyć Office 365 to najlepiej... A nawet nie powiedziałem, bo mamy sponsora. Poprosimy, różowy – wciśnij!

Ach...

[Ł. F] Ach...

[M. G] Ach...

[M. G] A sponsorem dzisiejszej audycji jest Dosia. Nie, co?

[Ł. F] ... jest Akademia Aplikacji. Słuchajcie, w opisie naszego odcinka znajdziecie kod zniżkowy, jeżeli chcecie samodzielnie zbudować aplikację w Power Apps, to mamy świetny kurs przetestowany przez wiele, wiele osób i no oczywiście nie jestem obiektywny, bo jestem autorem tego kursu wraz z Michałem.

[Ł. Z] Ale już zupełnie, zupełnie serio. Mamy bardzo dużo mega inspirujących przykładów po tym kursie. Ludzi, którzy naprawdę od poziomu takiego zupełnie, zupełnie zero byli w stanie pchnąć swój biznes.

[Ł. F] Chcesz naprawić pralkę? Power Apps! Nie działa mikrofalówka? Power Apps!

[M. G] Aż tak, to nie.

[Ł. Z] Wyciszamy.

[Ł. F] Chcesz zrobić lotnie balkon? Power Apps!

[M. G] Żegnaj! Ale to co ważne jest, żeby dodać to, że mając dostęp do kursu, dostaniecie dostęp do grupy i na tej grupie, te wszystkie przykłady są publikowane.

[Ł. F] Wkurza cię sąsiad? Power Apps!

[M. G] To ja go nie wyciszyłem do końca.

[Ł. F] Tak jest.

[Ł. Z] Dobra? Myślę, że czas na draft.

DRAFT

Jak w każdym drafcie, uczestnicy dzielą się pięcioma pomysłami przygotowanymi na ustalony wcześniej temat. Prelegenci mają za zadanie podać Top 5 najlepszych kandydatów na aplikację stworzoną w low-code, którą każda organizacja powinna wdrożyć. Aplikacja powinna być prosta, zrealizowana w około miesiąc. **Pierwsza aplikacja low-code:** Generator maili. Aplikacja tworząca spersonalizowane maile według wybranych danych klienta. (M. G) CRM (customer relationship management) – aplikacja pomagająca zarządzać relacjami z klientem. Jej funkcje byłyby spersonalizowane pod każdego klienta, stąd jest to projekt low-code, który można zrealizować w każdej firmie. (Ł. Z) Proces księgowy, ze szczególnym uwzględnieniem śledzenia kosztów. (Ł. F) **Druga aplikacja low-code:** Aplikacja do obiegu faktur - wystawianie faktur, wysyłanie faktur, zaznaczenie statusu faktur. (Ł. F) Wewnętrzny system zamówień - wewnętrzne „Allegro” - umożliwiające śledzenie stanów magazynowych i wykonywanie zamówień z poziomu aplikacji. (M. G) Service desk. (Ł. Z) **Trzecia aplikacja low-code:** Rozliczanie i śledzenie czasu pracy. (Ł. F) E-obiegówka. Na rynku dostępne są gotowe rozwiązania, ale niespersonalizowane pod potrzeby firmy. Aplikacja pozwalałaby między innymi na rezerwację sal konferencyjnych, projektorów czy samochodów służbowych. (M. G) Intranet – newsletter pozwalający na szybki dostęp do nowych informacji w ramach wewnętrznej sieci. Szczególnie przydatne w czasach pracy hybrydowej. (Ł. Z) **Czwarta aplikacja low-code:** Wspomniana wcześniej sieć Intranet umożliwiłaby dostęp do aplikacji dostępnych w danej firmie. (Ł. Z) Aplikacja low-code regulująca wnioski urlopowe. (Ł. F) Aplikacja pozwalająca na rezerwację szkoleń. Pracownik mógłby zarezerwować daną datę określonego kursu, co umożliwiłoby dodatkowo tworzenie grup startu po zadeklarowaniu odpowiedniej liczby chętnych. (M. G) **Piąta aplikacja low-code:** Skrzynka pomysłów. (Ł. Z) System rekomendacji pracowniczych, pozwalający na ocenienie danego pracownika. (M. G) Hot desking – rezerwacja biurek (przestrzeni biurowej) przez pojedynczych pracowników. Aplikacja byłaby szczególnie pomocna dla osób szukających tradycyjnej przestrzeni do pracy.

[Ł. F] Low-code, low-code. Draft wokół low-code będzie następujący: trochę bez szydery, nieprześmiewczy. Spróbujmy wskazać, waszym zdaniem, takie Top 5 - ale wiecie, to ma być naprawdę, takie najlepsze waszym zdaniem - najlepszych kandydatów na aplikację zrobioną w low-code, którą każda organizacja powinna wdrożyć. Tylko tutaj takie założenia, co oznacza low-code? Do wyklikania w miesiąc przez jedną osobę, czyli nie wymyślamy tam ERP - wiecie - i systemu do zarządzania wszechświatem. Proste rozwiązania, takie low hanging fruits - mówiąc po staropolsku

- po które warto sięgnąć, jeżeli zorientowałem się na przykład, że „Och mam Power Apps, to co z tym zrobić?” I nie powtarzamy swoich.

[Ł. Z] Tak, tu jest duża szansa właśnie.

[M. G] Tak, to kto zaczyna?

[Ł. F] No to powinniśmy losować, bo to w sumie ważne kto zaczyna.

[M. G] Dobra to ja mogę zaczynać. Pewnie i tak nie macie mojego. No dobrze, to moja pierwsza propozycja, to jest bardzo ciekawy przykład, który namierzyłem na projekcie jednego z klientów. Mianowicie jest to duża organizacja, która zajmuje się kwestiami prawniczymi i ubezpieczeniowymi i oni bardzo często wysyłają do swoich klientów podobnej treści wiadomości, czy to są wiadomości reklamacyjne, czy są z przeprosinami, czy to są jakieś wnioski, prośby o pogonienie, cokolwiek. I są szablony, które za każdym razem trzeba uzupełnić. Oni wcześniej mieli te szablony podpisywane w różnych plikach wordowych, ale dodatkowo zawsze było tak, że taki plik Microsoft Word po otwarciu wymaga jakiejś edycji, bo na przykład ten klient akurat nie miał tego, jakiejś tam jednej opcji albo drugiej opcji. Tak więc ta firma miała fantastyczną - znaczy miała, zbudowaliśmy - świetne rozwiązanie, które pozwalało na generowanie szablonów w taki sposób, że otwierają się aplikacje i na przykład mówiłeś, że reklamacja i czy klient ma to? Nie ma tego, tamto ma. A jak się to nazywa to? To tak i tak. I jaki ma tam samochód z jakimi rejestracjami? Klikacie „wygeneruj” i pojawia się copy paste, w sensie tekst, który się brało, kopiowało, wklejało do maila i gotowe. Wszystko tam było.

[Ł. F] Generator maili, to nie mogłeś tak krótko?

[M. G] Generator, generator wiadomości. Tak, mogłem tak krótko, ale według mnie to ważne, żeby ten kontekst cały zarysować.

[Ł. Z] Dobrze to pierwsza aplikacja dla mnie, którą firmie bym postawił w miesiąc na szybko, aby się mega przydała to CRM.

[Ł. F] Hm... Tak po prostu?

[M. G] Ale oczywista oczywistość.

[Ł. F] No tak no, ale to no sumie coś w tym jest.

[Ł. Z] Myślę o podstawowym procesie.

[M. G] Ale, ale poczekaj, a nie lepiej byłoby na przykład wziąć gotowego? Bo akurat generator maili ma – taki CRM jest gotowy.

[Ł. Z] Zakładam, że że nie robimy tych wyborów. No, są generatory.

[M. G] No to są generatory maili, ale taki żebyś miał pod swoją organizację skonfigurowany? Bo to chodzi o coś takiego bardzo customizowanego pod ciebie, a CRM są na rynku takie darmowe shoty.

[Ł. Z] No właśnie, ale im więcej ich jest, to tylko pokazuje, że ich potrzeby są unikalne, więc można jak najbardziej moim zdaniem. Bo nie wiem, może źle zrozumiałem, że mamy te najbardziej znane wybrać, więc i tak one będą – tak, ja to rozumiem. Wszystkie, które podamy, to prawdopodobnie znajdę ze trzech konkurentów na rynku

i mogę kupić to. Myślę o takim, którym bym był sam w stanie wyklikać według swoich potrzeb. Tak do tego podszedł.

[Ł. F] A nie mam gotowego rozwiązania?

[Ł. Z] Nie mam gotowego. Znaczący nie wybieram gotowego, bo pewnie ono jest.

[M. G] Ok.

[Ł. F] No dobra, to ja na pierwszy rzut zrobiłbym rozliczanie różnego rodzaju wydatków, czyli jakiś proces księgowy.

[Ł. Z] A opcja „całościowy”, na przykład delegacje?

[Ł. F] No właśnie delegacje tak, bo to jest bardzo duże.

[Ł. Z] Niestety jak powiesz, że procesy rozliczania księgowy to byś się załapał na dłużej niż w miesiąc.

[Ł. F] Rozliczanie kosztów, po prostu.

[M. G] Jakich, kosztów delegacji?

[Ł. F] Jakichkolwiek.

[M. G] Ale zabierasz mu pozycję albo nie zabierasz.

[Ł. F] Wszelkich.

[M. G] Czyli delegacji też?

[Ł. F] Czyli proces pod tytułem „wydaje pieniądze firmowe, muszę się z tego rozliczyć”. ok, dobra, no dobra, a czy to jest śledzenie kosztów...

[M. G] Śledzenie kosztów. No to zabrałeś mi pozycję. Do diabła! Druga teraz twoja.

[Ł. Z] No to dla mnie takim naprawdę sztampowym, ale często spotykany case to różnego rodzaju service deski i systemy aplikatowe.

[Ł. F] To ja idę, kontynuuję swoją myśl CRMa przechodzę do obiegu faktur, wystawianie faktur, wysyłanie faktur, zaznaczenie statusu faktur.

[M. G] To jest duże, to nie wiem czy w miesiąc da radę?

[Ł. F] W miesiąc spokojnie, bo jakby wyglądał ten system. Potrzebuję mieć szablon tej faktury. To jest 2 dni roboty, no umówmy się maks, potrzebuję mieć wpinkę do danych do CRMa i zrobienie tej faktury, a potem zmiany status quo.

[Ł. Z] Spryciarz, on po piątym będzie miał RPa.

[M. G] Teraz ja? Więc ja bym zrobił po pierwsze zarządzanie zasobami, czyli jestem w stanie na przykład zarezerwować czy zamówić z magazynu jakieś zasoby. Ja się specjalnie nie wysilałem, bo patrzę na projekty, w których brałem udział i które rzeczywiście widziałem, że wdrażały się przynajmniej raz, więc zamówienie, takie wewnętrzne allegro powiedzmy. Zamówienia pach! Zamawiam sprzęt. I to są rzeczy do zrobienia w miesiąc tyle trwa projekt. To jest jedna rzecz, a druga rzecz to karty obiegowe, coś co w szczególności w dużych organizacjach trzeba się tam rozliczyć ze sprzętu, który mieliśmy. Jakies tam stemple z różnych działów.

[Ł. Z] Taka e-obiegówka.

[Ł. F] Nachodzić się trzeba za cholereę po prostu.

[M. G] No tak tak. E-obiegówkę można bardzo łatwo, bo jak bierzemy gotowe rozwiązania z rynku, to jest znowu case, że na rynku są podobne rozwiązanie, tylko wtedy trzeba zapłacić licencję od użytkownika, coś tam coś tam.

[Ł. Z] Bo często to są moduły jakieś większe.

[M. G] I moduły, a tutaj po prostu możemy sobie zbudować taką aplikację. Ona nic nie kosztuje, elo 3 2 0 ciach.

[Ł. Z] Nie zrozumiałem na końcu.

[Ł. F] EE to Michał jest raperem.

[M. G] Człowiek nie czuje, kiedy rapuje.

[Ł. F] Też o zasobach, rezerwacja zasobów?

[Ł. Z] Czyli salki, samochody, projektory? Wszystko siedzi na kalendarzach. No wydaje mi się, że bez tego ani rusz.

[M. G] Tak, rezerwowanie zasobów.

[Ł. F] To ja mam dwa takie naprawdę, na pewno mieliście je na liście. Więc zabieram od razu pierwszy to rozliczanie i śledzenie czasu pracy.

[Ł. Z] Kurteczka.

[Ł. F] Kontrowersyjny temat, ale sami przyznacie, że to jest przydatne. No i wnioski urlopowe.

[Ł. Z] Pierwsze na mojej liście.

[Ł. F] To czemu nie zacząłeś od tego?

[Ł. Z] CRM mi się spodobał w międzyczasie bardziej, ale miałem pierwszy. To jest teraz moje, moje.

[Ł. F] Tak ostatnie.

[Ł. Z] Chyba nie, przedostatnie. Leł to było tak.

[M. G] To ty masz jeszcze, ty masz jeszcze dwa.

[Ł. Z] A tak, dokładnie. To ja pomyślałem o tym, żebym zrobił sobie intranet, który byłby rozumiany jako...

[M. G] Pamiętaj, że masz miesiąc, ale kalendarzowy.

[Ł. Z] Tak, ale na którym był chciał mieć tylko w sumie dwie rzeczy: hub aplikacyjny do tego wszystkiego o czym mówimy, żebym mógł się przekierować i chciałbym mieć aktualności.

[Ł. F] Oj, to jest nie głupie, bo my się mieliśmy z Michałem na Automate.

[M. G] To jest taki newsletter, taki czytnik newsów.

[Ł. Z] Takich hub, to jest moje miejsce, z którego zaczynam.

[M. G] No spoko, no ok.

[Ł. Z] No myślę, że w czasach hybrydowej pracy.

[M. G] Dobra, to teraz ja mam dwa ostatnie, pierwsze to jest zarządzanie szkoleniami wewnętrznymi. Jestem pracownikiem, mam do wyboru jakieś szkolenia, chcę się zarezerwować na jakąś tam konkretną datę, dane szkolenie - ciach. Tam się wpisuje do grupy grupa startu jak ma ileś tam osób. Dopiero więc coś takiego.

[Ł. Z] A naszym sponsorem jest akademia aplikacji.

[Ł. F] Ale idealnie to wcisnąć, bo chciałem powiedzieć, że na Akademię Aplikacji łamane przez webinary jest nagranie ze spotkania „jak zbudować aplikację w 30 minut” i ja robiłem taką aplikację.

[M. G] No, no właśnie, to nawet 30 minut, a nie w tydzień, w miesiąc. No dobra, a druga rzecz to jest system ocen pracowniczych. Kiedy możecie sobie ocenić czy wystawić komuś jakąś rekomendację, tego typu rzeczy.

[Ł. Z] PDRy to były?

[M. G] A nie NPS?

[Ł. Z] A ja ostatnie - ciekawe czy was zaskoczę - liczyłem na to, że po prostu wyjedziemy z tymi nastoma pomysłami. Skrzynkę pomysłów, bym chciał. Fajne, bo to jest rzadziej pomyślane w pierwszej kolejności, a ja uważam, że tam by dużo fajnych rzeczy by się pojawiło.

[Ł. F] Swoją drogą wiele ciekawych systemów tego typu robiliśmy.

[Ł. Z] No i tak przypomniało mi się wczoraj, kiedy się kładłem spać.

[Ł. F] Niektóre skomplikowane, niektóre prostsze.

[Ł. Z] Myślę, że przy tym jak już wiemy, jak to wygląda, to też jest szybko do zrobienia.

[Ł. F] Moja ostatnia propozycja, na czasie mocno a propos reorganizacji pracy i pracy hybrydowej, też zauważam dużo tego typu rozwiązań powstaje do hot deskingu -rezerwacji biurka.

[M. G] Ale to może być skomplikowane, bo co, jeżeli te biurka na przykład zmieniają swoją lokalizację.

[Ł. Z] Not to jest w aplikacji już u Fali, to trzeba zobaczyć.

[Ł. F] Biurko zmienia lokalizację?

[M. G] No bo to są hot deski.

[Ł. F] To ty zmieniasz lokalizacje biurka. Są w tych samych miejscach na ogół.

[Ł. Z] To nie jest jak na weselu z tymi krzesłami. Nie, nie.

[Ł. Z] No to co Fala, zepniesz to jakąś ładną kłamrą? Kończymy, robimy?

[M. G] Do widzenia!

[Ł. F] Róbcie low-code i do następnego razu! Cześć! To był draft rozmowy. Jeśli wam się podobało, subskrybujcie nasz podcast i nie zapomnijcie wystawić ocenę. To dla nas ważna rzecz motywuje do dalszej pracy i pozwala rozwijać ten podcast.

[M. G] A na facebooku założyliśmy specjalną grupę, na której możecie komentować nasze przemyślenia oraz wybory w draftach.

[Ł. Z] Pamiętajcie też, że poniżej znajdziecie opis tego odcinka, a w nim wszystkie przydatne linki.

[Ł. Z] To był draft rozmowy. Jeśli wam się podobało, subskrybujcie nasz podcast i nie zapomnijcie wystawić ocenę. To dla nas ważna rzecz. Motywuje do dalszej pracy i pozwala rozwijać ten podcast.

[M. G] Na facebooku założyliśmy specjalną grupę, na której możecie komentować nasze przemyślenia oraz wybory w draftach.

[Ł. F] Pamiętajcie też, że poniżej znajdziecie opis tego odcinka, a w nim wszystkie przydatne linki.