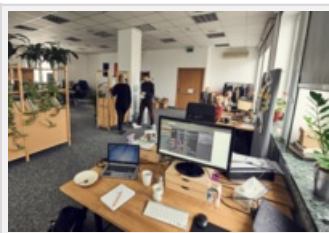


Najefektywniejszy zespół programistów

2021-02-21



Opiszę tu moje doświadczenia z rodzajami **zespołów tworzących oprogramowanie**. W ciągu ostatnich ponad 20 lat współuczestniczyłem w kilku typach zespołów programistycznych i były to głównie:

1. Zespół z technicznym Dev Leadem (i jednocześnie Project Managerem)

- kilku programistów programuje
- Dev Lead programuje
- Dev Lead tworzy i przydziela zadania programistom
- Dev Lead robi Code Review i merguje / wgrzywa zmiany do systemu
- Dev Lead rozwiązuje problemy nietechniczne (rekrutacja, urlopy, formalności itp)
- Dev Lead rozmawia z **Klientem (Product Ownerem)** odnośnie funkcjonalności i projektu
- Dev Lead jest odpowiedzialny za testy systemu (ew zespół testowy)
- Dev Lead jest odpowiedzialny za support systemu (błędy, zgłoszenia)

2. Zespół z nietechnicznym Project Managerem

- kilku programistów programuje (i ew. robią nawzajem code review, wgrzywają zmiany do systemu)
- Project Manager tworzy i przydziela zadanie
- Project Manager rozmawia z **Klientem (Product Ownerem)** odnośnie funkcjonalności i projektu
- Project Manager rozwiązuje problemy nietechniczne (rekrutacja, urlopy, formalności itp)
- Project Manager jest odpowiedzialny za testy systemu (ew zespół testowy)
- Project Manager jest odpowiedzialny za support systemu (błędy, zgłoszenia)

3. Zespół z technicznym Dev Leadem i Project Managerem

- kilku programistów programuje
- Dev Lead programuje
- Dev Lead tworzy i przydziela zadania programistom
- Dev Lead robi Code Review i merguje / wgrzywa zmiany do systemu
- Project Manager ustala z **Dev Leadem** zadania do wykonania i ew. kwestie urlopów / podwyżek
- Project Manager rozmawia z **Klientem (Product Ownerem)** odnośnie funkcjonalności i projektu
- Project Manager rozwiązuje problemy nietechniczne (rekrutacja, urlopy, formalności itp)
- Project Manager jest odpowiedzialny za testy systemu (ew zespół testowy)
- Project Manager jest odpowiedzialny za support systemu (błędy, zgłoszenia)

Zdecydowanie najefektywniejszy jest zespół nr. 3. **Zespół z technicznym Dev Leadem i Project Managerem** Oparte jest to zarówno na moich doświadczeniach, jak też na rozmowach z osobami pracującymi w innych firmach IT (w tym z programistami z największych światowych firm technologicznych).

Największy problem z 2 typem zespołu (**Zespół z nietechnicznym Project Managerem**) jest taki, że przy większych zadaniach PM nie jest w stanie określić czasu wykonania i podzielić zadania na mniejsze podzadania (najlepiej tak aby np. każde z tych mniejszych podzadań dało się wykonać w kilka dni). Dodatkowo jeśli nie ma DevLeada, wtedy często nie ma osoby, która czuje się odpowiedzialna za spójność stylu programowania, używane narzędzia, ułatwienia pracy wszystkich programistów itp.

Jeśli zadania programistyczne jest dłuższe niż kilka dni to bardzo często nie da się go dokładnie oszacować i można je wykonać zarówno w **kilka dni**, w **kilka tygodni** lub w **kilka miesięcy**. Różnice w podejściu do rozwiązania problemu mogą być często ocenione tylko przez osoby techniczne - osoby nietechniczne nie mogą konstruktywnie zaproponować, gdy np. programista stwierdzi, że aby wykonać dane zadanie trzeba od nowa przepisać inne części systemu. Ważne jest więc, aby zadania rozdzielała osoba, która może zdecydować jak technicznie powinny być one rozwiązane, oraz jeśli są duże to była w stanie podzielić je na zadanie mniejsze (np. maksymalnie kilkudniowe) i te małe zadania przydzielać programistom. Project Manager bez doświadczenia programistycznego nie będzie mógł tego zrobić.

Dużo efektywniejszy niż zespół nr. 2 jest zespół nr. 1 **Zespół z technicznym Dev Leadem** - pomimo, że Dev Lead może tracić dużo czasu na rozmowy / e-maile / telefony z klientami. Więc jeśli jest możliwość dołożenia do takiego

zespołu dodatkowego nietechnicznego Project Managera (i tym samym stworzyć **Zespół z technicznym Dev Leadem i Project Managerem**) to zawsze warto to zrobić.

Innym problemem jest znalezienie **Dev Leada** do projektu. Rekrutacja programistów jest bardzo trudna, a Dev Lead powinien być przede wszystkim bardzo dobrym i doświadczonym programistą. Jeśli jednak odpowiednio podzielimy obowiązki Dev Leada, tak aby mógł on maksymalnie skupić się na kwestiach technicznych i jeśli jak najwięcej nietechnicznych zadań przejmie **Project Manager** to łatwiej będzie nam znaleźć osobę na **stanowisko Dev Leada**. Zwłaszcza, że rola Dev Leada jest bardzo **rozwijająca technologicznie**, gdyż projektuje on architekturę systemu oraz nowych funkcjonalności, czyli ma bardzo duży wpływ na to, jak technicznie będzie rozwijał się projekt.

Oczywiście, ważne jest też, aby **Dev Lead** miał w zespole jak najlepszych programistów, którzy albo już dobrze programują, albo też szybko się uczą i zapowiada się, że niedługo będą "dobrze" programowali. Jeśli Dev Lead będzie miał "zły" zespół programistów i jego zadaniem będzie głównie ciągle poprawianie kodu innych osób, to cały zespół nie będzie efektywny, a sam Dev Lead szybko zmęczy się swoją rolą. Dev Lead powinien więc mieć wpływ na to kto pracuje jest w jego zespole.

Kolejną kwestią jest wdrożenie **Dev Leada** do pełnienia swojej funkcji. Potrafi on bardzo dobrze programować i efektywnie rozwiązywać problemy, ale często nie miał doświadczenia w delegowaniu zadań. Najłatwiej jest, gdy wcześniej pracował on jako **programista** w zespole innego Dev Leada i dodatkowo uważa, że projekt był prowadzony dobrze i chce przejąć używane tam praktyki. Jeśli jednak kandydat na Dev Leada nie ma doświadczenia poza programistycznego i nie ma dobrego przykładu, na którym mógłby się wzorować, to warto jest poważnie zastanowić się jak najłatwiej wdrożyć go do nowych obowiązków (kiedyś postaram się o tym napisać, np o naszych narzędziach do zarządzania zadaniami).

W mniejszych zespołach **Dev Lead** może głównie programować, natomiast przy większych zespołach i projektach więcej czasu może zajmować mu nadzór nad architekturą, zadaniami i kodem innych osób. Może się też okazać, że na pewnym etapie projektu niezbędny będzie osobny zespół np. DevOps. Podobnie **Project Manager** w małych zespołach i przy małej liczbie klientów może więcej rzeczy wykonywać samodzielnie, natomiast wraz ze wzrostem projektu może potrzebować wydzielić i nadzorować osobne zespoły np. do Supportu. Czasem też w naszych zespołach Project Manager dodatkowo wykonywał zadania z zakresu Marketingu lub pełnił częściowo rolę Product Ownera i wtedy nazywaliśmy to łączone stanowisko **Product Managerem**.