

Falcon V Systems Case Study

LeSS w praktyce



Jak skalować Scruma bez utraty jego zalet?

Scrum został opracowany jako sposób na efektywną pracę w niewielkich zespołach. W końcu jednak dochodzisz do tego momentu, w którym chcesz skalować swój biznes. Jak w ciągu kilku miesięcy urosnąć kilkukrotnie i zachować dużą elastyczność działania? Poznaj historię polskiej firmy Falcon V Systems i sprawdź, jak Large-Scale Scrum (LeSS) pomógł im zbudować adaptacyjną organizację i eksperymentalnie odkrywać najlepsze rozwiązania.

O Falcon V Systems

Falcon V Systems¹ został założony w 2018 roku przez Liberty Global², globalnego operatora telekomunikacyjnego oraz Grupę VECTOR³, która od ponad 30 lat projektuje, wdraża i integruje najnowocześniejsze systemy na rynku technologii komunikacyjno-informacyjnych (ICT). Celem tej współpracy było stworzenie firmy oferującej rozwiązania software'owe dla operatorów telekomunikacyjnych, przyspieszając ich migrację w kierunku integracji technologii dla sieci dostępowych nowej generacji. Dzięki temu mieli oni lepiej i szybciej dostarczać swoje usługi do klientów końcowych, zapewniając im szybki i nieprzerwany dostęp do Internetu. Idea produktu była na tyle rewolucyjna, że strona techniczna, czyli sposób zaspokojenia potrzeb użytkowników, nie pozwalała na zdefiniowanie wszystkich szczegółów z góry oraz wymagała odpowiedniej ilości badań i eksperymentów. Ich wyniki miały generować nowe pomysły i możliwości, jak i kolejne wyzwania do rozwiązania. W takiej sytuacji założyciele zdecydowali się na adaptacyjne podejście do tworzenia nowego produktu, aby tworzoną wiedzę przekładać na zwiększenie wartości dla klienta.



¹ <https://falconvsystems.com/>

² <https://www.libertyglobal.com/>

³ <https://vector.net/>

Wiele zespołów pracujących nad jednym produktem

Pierwsze badania firma przeprowadzała przy iteracyjnym podejściu. Siedmioosobowy zespół korzystał z frameworku Scrum⁴, żeby dostarczać kolejne wersje prototypu produktu w krótkich cyklach zwanych Sprintami. Apetyt rósł w miarę jedzenia, więc osiągnięte rezultaty wkrótce zachęciły inwestorów do podjęcia decyzji o gwałtownym powiększeniu zespołu pracującego nad produktem. W krótkim czasie firma miała urosnąć do 50-ciu osób, co oczywiście musiało wpłynąć na sposób jej pracy.

Scrum bazuje na niewielkich, kilkusobowych zespołach ludzi ściśle współpracujących ze sobą na co dzień, dlatego próba zastosowania go przy kilkudziesięciu osobach mogłaby prowadzić do ograniczenia produktywności, ale przede wszystkim, założonej na początku adaptacyjności.

Falcon V Systems zaczął poszukiwać rozwiązania umożliwiającego pracę w kilku podzespołach (zwanymi wewnątrz firmy Drużynami). Z dostępnych opcji wybór padł na LeSS (Large-Scale Scrum)⁵, którego twórcy z jednej strony starali się zachować adaptacyjny charakter Scruma, a z drugiej stworzyli bazę zasad, wytycznych i eksperymentów, ułatwiających jego wdrożenie w firmach różnej wielkości, tworzących różne produkty i działających na różnych rynkach. Przeszkolenie, najpierw zarządu, a następnie wszystkich pracowników, pozwoliło firmie upewnić się, że jest to optymalne rozwiązanie.

Ponad cztery lata doświadczeń pozwoliły firmie dowiedzieć się jak:

- A.** Podejście eksperymentalne pozwala dobrać narzędzia i techniki działania.
- B.** Definicja produktu wpływa na adaptacyjność firmy.
- C.** Stworzyć strukturę organizacji wspierającą ciągłe uczenie się inżynierów.
- D.** Brak formalnych koordynatorów buduje silną współpracę między zespołami.
- E.** Zbudować wspólne zrozumienie nad czym firma będzie pracować.
- F.** Praktyki inżynierskie wspierają równoczesną pracę wielu zespołów nad jednym produktem.
- G.** Zapewnić utrzymanie produktu.

⁴ Scrum to model pracy bazujący na iteracyjnym dostarczaniu wartości <https://procognita.pl/scrum>
⁵ Large-Scale Scrum (LeSS) to bazujący na Scrumie model pracy pozwalający wielu zespołom iteracyjnie rozwijać produkt <https://procognita.pl/less>

A. Podejście eksperymentalne

LeSS zawiera niewielką liczbę reguł, jednak ich wdrożenie niesie ze sobą istotne konsekwencje dla firmy. Po pierwsze reguły te wpływają znacząco na większość istotnych elementów zarządzania firmą, zaczynając od struktury organizacji, poprzez sposób wykonywania pracy, a kończąc na rekrutacji i rozwoju pracowników. Wdrażanie ich na siłę, bez wytłumaczenia stojących za nimi przyczyn (zasad) będzie powodowało silny opór zarówno pracowników, jak i kadry managerskiej. Próba wprowadzenia zmiany z uzasadnieniem „bo w LeSSie jest tak” albo „bo Scrum tak mówi”, zagwarantuje ciągłą dyskusję i ciągłe kwestionowanie wprowadzanego podejścia. Przeszkolenie wszystkich pracowników z pewnością pomogło zaadresować część obaw, jednak nie zastąpiło doświadczenia, które często zdobywali na własnych błędach.

Po drugie, ograniczona ilość reguł powoduje, że LeSS wymaga przeprowadzenia wielu eksperymentów. Wytyczne LeSSa zwykle nie odpowiadają na pytania jak coś zrobić – na przykład jak efektywnie rozpropagować na cały zespół specjalistyczną wiedzę domenową posiadaną na początku jedynie przez kilka osób. Do tego Drużyny musiały dojść same. Oznaczało to porzucenie podejścia „to się nie uda” i „w poprzedniej firmie już tego próbowaliśmy” i częstsze używanie słowa „spróbujmy”. Wymagało to również zaangażowania pracowników poprzez jasne określenie i zakomunikowanie celu eksperymentu, czasu jego trwania, oraz po czym poznają, że osiągnęli oczekiwane rezultaty.

Bardzo pomocne w tym okazały się cykliczne spotkania nazywane „Overall Retro”, na których reprezentanci Drużyn, Scrum Masterzy i managerowie:

- **Omawiają problemy i pomysły przyniesione przez Drużyny,**
- **definiują nowe eksperymenty dookoła procesu, produktu jak i praktyk inżynierskich,**
- **analizują osiągnięte rezultaty.**

Jednym z pierwszych eksperymentów, był warsztat samo-definiowania się zespołów (Team Self-Designing Workshop)⁶. Po podjęciu decyzji biznesowej o skalowaniu firmy, dwie istniejące wtedy Drużyny gwałtownie urosły i mając po kilkanaście osób, zaczęły być za duże, żeby pracować efektywnie. Rozkład potrzebnych kompetencji też nie wydawał się optymalny, a rozproszenie na dwie lokalizacje utrudniało bliską, codzienną współpracę twarzą w twarz. W trakcie jednego z kwartalnych zjazdów firmowych czwórka Scrum Masterów przeprowadziła warsztat, na którym pracownicy sami przypisali się do nowych zespołów. W rezultacie, uzyskano pięć niewielkich, interdyscyplinarnych Drużyn, z których każda w całości znajdowała się w jednej z dwóch lokalizacji. Choć struktura trochę ewoluowała przez następne lata, to wypracowane podejście pozostawiło decyzyjność i odpowiedzialność za nią na poziomie inżynierów.

⁶ Dokładniejszy opis warsztatu można znaleźć na <https://less.works/blog/2018/12/27/how-to-form-teams-a-story-of-self-designing-teams.html>

B. Ile firma ma produktów?

Tak jak zostało wspomniane wcześniej, podejścia zwinne, takie jak Scrum czy LeSS bazują na pracy małych, interdyscyplinarnych zespołów potrafiących iteracyjnie dostarczać kolejne wersje produktu (tzw. Inkreментy). Jednak to, co firmy definiują jako „produkty” często różni się od tego, jak postrzegają je klienci.

Jednym z głównych założeń budowania rozwiązań w Falcon V Systems była interoperacyjność, która umożliwiała komunikację z innymi urządzeniami w ramach jednego, otwartego ekosystemu (Open DAA)⁷. W zależności od architektury sieci, operator telekomunikacyjny mógł wybrać produkty dostosowane do technologii, w której chce się dalej rozwijać. Tym samym, definicja „produktu” dla każdego klienta mogła być inna.

Jednocześnie każda Drużyna potrzebowała wiedzieć, co wchodzi w zakres jej prac i Inkrementu, czyli co jest jej „produktem”. Z jednej strony LeSS rekomenduje szeroką definicję „produktu”, to znaczy, aby wszystkie Drużyny wspólnie tworzyły jeden Inkrement. Z drugiej strony, w krótkim odstępie czasu startup zaczął pracować nad dwoma odrębnymi rozwiązaniami dla dwóch różnych klientów, co potęgowało uczucie, że traktowanie ich jako jednego produktu jest nieefektywne. Firma zdecydowała się więc zdefiniować dwa obszary (tzw. Area) - Service Orchestration (SO) oraz Service Delivery (SD) i przypisać do nich odpowiednio jedną i cztery Drużyny.

Warto tu podkreślić, że wąskie definicje „produktów” i specjalizacje zespołów dookoła technologii (np. C++, Java), komponentów (Backend, Frontend), czy domeny biznesowej (płatności, rejestracja nowych użytkowników) są bardzo popularnym podejściem na rynku. Na pierwszy rzut oka ma to sens, bo znacząco ułatwia pracę zespołowi, ograniczając ilość potrzebnych informacji – np. nie muszą one znać szczegółowo planów rozwoju całej aplikacji (tzw. Backlog), a jedynie te elementy, które dotyczą ich obszaru. Zmniejsza to też potrzebę znajomości całego systemu i posiadania wielu kompetencji. Inaczej mówiąc, w przypadku wystąpienia problemu z określonej dziedziny, mamy dedykowany zespół, który się nim zajmie. Jednak w długiej perspektywie zmniejsza to adaptacyjność firmy.

Właśnie z takim wyzwaniem musiał zmierzyć się zespół Falcon V Systems decydując się na pracę w dwóch obszarach. Jedna Drużyna wyspecjalizowana w obszarze pierwszym (SO) miała znacząco ułatwioną pracę w porównaniu do pozostałych czterech Drużyn z obszaru drugiego (SD), które współpracowały nad jednym wspólnym Inkrementem (i jednym Backlogiem). Tutaj inżynierowie wyraźnie odczuli wyzwania skali. Niezbędna okazała się częsta synchronizacja, zarówno na poziomie komunikacji między Drużynami, jak i praktyk inżynierskich – chcąc dostarczyć jeden Inkrement musieli unikać „merge festival”, w którym wszystkie zmiany są wprowadzane pod koniec iteracji. Również prowadzenie spotkań przygotowujących elementy Backlogu na kolejne Sprints przy kilkadziesiąt osób okazało się dużo bardziej bolesne niż w przypadku jednej Drużyny z obszaru pierwszego. Zwłaszcza w sytuacji nagle wymuszonej pracy zdalnej, gdyż po półtora roku od utworzenia firmy zastała ją pandemia.

⁷ Więcej na temat ekosystemu Open DAA można znaleźć na <https://falconvsystems.com/demystifying-open-daa-approach/>

Pomimo tego zespół starał się zachować spójne procesy, środowiska pracy, narzędzia, sposób działania i kulturę organizacyjną w obu obszarach. Wymagało to pewnego wysiłku, bo wiele problemów większego obszaru drugiego (SD) nie występowało w ogóle w obszarze pierwszym (SO). Różnice zdań najmocniej wybrzmiewały w trakcie Overall Retro, gdzie trwały zagorzałe dyskusje na temat istotności podejmowanych wyzwań i potencjalnych możliwości ich rozwiązania na poziomie całej organizacji.

Zysk ze spójnych procesów i unikania wąskiej specjalizacji Drużyn stał się jasny w momencie pozyskania przez firmę intratnego kontraktu umożliwiającego wejście na nowy rynek. Dzięki zdobytemu doświadczeniu we wspólnej pracy wielu zespołów nad jednym produktem, wszystkie Drużyny z obszaru drugiego (SD) zostały sprawnie przeniesione do obszaru pierwszego (SO). Wypracowany model pracy umożliwił więc adaptację na poziomie całej organizacji i skupienie wysiłków na tym, co potencjalnie mogło przynieść największą wartość. Tym samym zrezygnowano z podziałów i zgodnie z sugestią LeSSa firma powróciła do pracy nad jednym dużym produktem.



„Szeroka definicja produktu daje dużą oszczędność ze względu na jeden proces, większy porządek poprzez jeden Backlog, bez próby osiągnięcia zbyt wielu celów naraz oraz jedną osobą decyzyjną – Product Ownera, odpowiedzialną za zysk z podejmowanej inwestycji. To czego się nauczyliśmy, to fakt, że w przyszłości chcąc rozpocząć pracę w nowym obszarze produktu, potrzebujemy jedynie zmniejszyć tempo rozwoju innego.”
– **mówi Michał Jakubas, Scrum Master w Falcon V Systems.**

„Absolutnie nie chcemy tutaj zasugerować, że takie zmiany strategii są trywialne. Musieliśmy zmierzyć się z problemami wynikającymi z potrzeby zdobycia nowych kompetencji i wiedzy domowej przez Drużyny uczące się dopiero nowego dla nich obszaru. Ostatecznie jednak uchroniło nas to od wywierania presji na tę nieliczną grupę, która jako jedyna znałaby się na krytycznym dla nas obszarze, co mogłoby prowadzić do produktu o niskiej jakości i wypalenia ludzi.”
– **dodaje Michał.**

C. Struktura organizacji - uczące się zespoły

Dążenie do większej adaptacyjności na poziomie organizacji ma również istotne konsekwencje dla kompetencji, których firma oczekuje od swoich pracowników. Wąskie specjalizacje, wspomniane powyżej, dają poczucie pełnego wykorzystania kompetencji ludzi, ale często uniemożliwiają pracę nad tematami o najwyższym priorytecie.

Falcon V Systems przekonał się o tym tworząc w pewnym momencie zespół specjalizujący się w zaawansowanych rozwiązaniach w języku C++. Początkowe rezultaty były bardzo obiecujące, bo wspomniana Drużyna pracowała szybciej niż gdyby te kompetencje zostały rozproszone po całej organizacji. Jednak ponowna zmiana potrzeb spowodowała, że domena, nad którą pracowali okazała się nie być priorytetowa.

„Nauczyliśmy się, że wąskie specjalizacje Drużyn mogą przynieść rezultaty krótkoterminowo, ale nie zapewnią pełnej adaptacyjności całej organizacji i umiejętności szybkiego reagowania na zmieniające się realia biznesowe. Uwzględniamy to więc w naszym procesie rekrutacji, poszukując ludzi chcących się rozwijać w różnych obszarach i patrzących na całość rozwiązywanego problemu, a nie jedynie przez pryzmat posiadanych kompetencji.”

– mówi Małgorzata Górską, COO Falcon V Systems.



„Posiadanie wiedzy i kompetencji przez wiele osób i Drużyn zabezpiecza nas przed kryzysem, który mógłby nastąpić w momencie utraty jedyne go specjalisty z krytycznego obszaru aplikacji. Oczywiście nie jest tak, że każdy pracownik umie wszystko, a każda Drużyna jest tak samo przygotowana do pracy w dowolnym obszarze produktu. Traktujemy to jednak jako długotrwałą inwestycję – proces, który ma nam zapewnić większą adaptacyjność organizacji i odporność na nieprzewidywalne wydarzenia. Dodatkowo, możliwość pracy nad danym zagadnieniem przez inne Drużyny wymusza w pewien sposób powstawanie czytelniejszego, zrozumiałego dla wszystkich kodu oraz lepszej dokumentacji.”

– mówi Bartosz Kajut, Członek Zarządu Falcon V Systems.

D. Współpraca między zespołami

Drużyny potrzebują ze sobą codziennie współpracować. Ich członkowie nie mogą zrzucić odpowiedzialności za identyfikację ewentualnych zależności oraz rozwiązywanie konfliktów na swoich przełożonych, ale muszą się nauczyć również i tych kompetencji. Szczególnie od 2020 roku, kiedy to praca zdalna i brak bezpośredniego kontaktu mogły wpłynąć negatywnie na poziom zaufania między ludźmi, a tym samym na chęć poruszania na forum pojawiających się problemów. Oczywiście brak rozmów o konfliktach nie oznacza ich nieistnienia, a same dyskusje nie gwarantują ich rozwiązania.

W związku z tym zespół Falcon V Systems przeprowadził sporą liczbę eksperymentów, które miały poprawić współpracę wewnątrz Drużyn, jak i pomiędzy nimi. Spośród tych udanych warto wspomnieć stałe połączenie na Zoom ustawione dla każdej z Drużyn, które daje im poczucie pracy w jednym pomieszczeniu, tak bardzo rekomendowanej przez podejścia zwinne. Ułatwia to również „zeglądnięcie” do pokoju innej Drużyny w celu zadania pytania albo dołączenia do ich codziennego spotkania. Ten sposób pracy zyskał w firmie nawet własną nazwę – „co-located co-call teams”.

„Bardzo pomogło nam przeprowadzenie cyklu „miękkich” szkoleń z komunikacji, facylitacji, podejmowania decyzji czy przekazywania informacji zwrotnej. Pozwoliło to pracownikom zarówno na lepsze tłumaczenie celów, które chcą osiągnąć proponowanymi zmianami, ale też na efektywniejsze wypracowanie eksperymentów, które miały większą szansę na akceptację przez pozostałych. Jednocześnie nauczyliśmy się na Overall Retro nie przechodzić od razu do wdrażania zaproponowanych akcji, ale zaczynać od analizy całego systemu z wykorzystaniem modelowania systemowego.”

– **mówi Karolina Kędzierska, Scrum Master w Falcon V Systems.**

Kolejną techniką wspierającą zarówno współpracę, jak i propagację wiedzy między Drużynami jest „Traveler”, czyli wypożyczanie ludzi do innych zespołów. Celem takiej osoby nie jest samo wykonanie zadania, ale raczej przekazanie wiedzy i umiejętności. Również cele Sprintów obejmujące kilka Drużyn zachęcają do ścisłej współpracy i częstszego korzystania z Travelerów.

Warto również wspomnieć co kwartalne spotkania, które w trakcie pandemii były kontynuowane w formie zdalnej (wliczając w to lanie wosku na andrzejki w 2020, odpowiednio wcześniej dostarczonego kurierem). Połowa tych spotkań jest skupiona na wymianie wiedzy, a druga na integracji pracowników. Obecnie firma korzysta z techniki Open Space⁸, która umożliwia każdej osobie zgłoszenie istotnego dla niej tematu i udział w interesujących ją rozmowach.

8 Więcej na temat Open Space można przeczytać na <https://openspaceworld.org/wp2/what-is/>

E. Rafinacja Backlogu, czyli nad czym będą pracować Drużyny

Jednym z największych wyzwań, wielokrotnie pojawiającym się w trakcie naszych rozmów z pracownikami, było efektywne przeprowadzanie porządkowania Backlogu (nazywanego w firmie Rafinacją), czyli ustalanie zakresu prac na nadchodzące Sprints. Celem tego spotkania jest zapewnienie zrozumienia zbliżających się prac przez wszystkich w zespole. Ma ono więc chronić przed negatywnymi konsekwencjami alternatywnego modelu pracy, w którym to osobne grupy produktowców i analityków przygotowują szczegółowo opisane wymagania a programiści je implementują. Wybrane podejście adresuje ryzyko braku pełnego zrozumienia i zbyt wąskiego spojrzenia na rozwiązywany problem. Dzięki temu relatywnie szybko uzyskujemy optymalne rozwiązania i jednocześnie inżynierowie czują odpowiedzialność za zaadresowanie potrzeb klienta.

Firma od początku promowała przekazanie Drużynom pełnej odpowiedzialności za tworzone rozwiązania, począwszy od zrozumienia problemu, poprzez analizę techniczną, aż do implementacji. W związku z tym dążyła do zaangażowania w Rafinację wszystkich inżynierów.

„Chcemy, żeby wiedza na temat elementów Backlogu, które będziemy implementować w najbliższej przyszłości była dostępna dla wszystkich Drużyn, a nie tylko wybranej. Po pierwsze zwiększa to naszą adaptacyjność, maksymalnie opóźniając decyzję, która Drużyna czym się zajmie w danym Sprincie, a po drugie pozwala uniknąć nieoptymalnych rozwiązań, gdyby mniej doświadczeni pracownicy jako jedyni projektowali rozwiązanie, a następnie je implementowali. Z tych powodów dążymy do współpracy wszystkich Drużyn w trakcie Rafinacji.”

– **mówi Maciej Słociński, Scrum Master w Falcon V Systems.**

Jednak sama świadomość ważności tego spotkania, nie powodowała, że było ono bezbolesne. Szczególnie na początku, kiedy specjalistyczna wiedza domenowa była w posiadaniu jedynie kilku osób z pierwotnego, niewielkiego zespołu. Chcąc więc zwiększyć efektywność Rafinacji, zespół eksperymentował z różnymi podejściami do ich przeprowadzania. Próbował tworzyć grupy robocze skupiające ludzi z różnymi kompetencjami, a także testował różne sposoby dokumentowania wypracowanej informacji. W trakcie prac, inżynierowie odkryli trzy istotne wzorce, które pomagają obecnie w przygotowaniu Backlogu:

- Po pierwsze, celem Rafinacji nie jest stworzenie dokumentacji, czyli precyzyjne opisanie elementu Backlogu, a zbudowanie zrozumienia. Efektywne przekazanie wiedzy jest dużo istotniejsze niż samo wygenerowanie dokumentów.

- Dlatego przeprowadzają Rafinacje z wykorzystaniem techniki połącz-podziel-połącz (merge-diverge-merge), w której najpierw dzielą się elementami do przeanalizowania pomiędzy grupami roboczymi, potem pracują w podgrupach, a na końcu spotykają się ponownie, żeby omówić wnioski i obserwacje.
- Po trzecie, większość elementów powinna przejść przez Rafinację dwa razy. Z tego powodu pierwotne pomysły są najpierw poddawane „pre-Rafinacji”, w trakcie której niewielka grupa definiuje cel i wstępny zakres, a dopiero później, gdy taki element jest już gotowy do Rafinacji, uwzględnia się go na głównym spotkaniu.

„Nie oznacza to, że rozwiązaliśmy wszystkie bolączki naszych Rafinacji. Brak znajomości całego produktu przez pracownika będzie mu utrudniać aktywne uczestnictwo w spotkaniu. Częste zmiany celów biznesowych będą generować potrzebę zrozumienia nowego obszaru i szybkiego przygotowania nowych elementów Backlogu, tak żeby je można było zaimplementować w kolejnym Sprincie. Tym samym Rafinacje na poziomie całego produktu są formą kompromisu, pomiędzy łatwością wąskiej specjalizacji w jednej domenie, a adaptacyjnym reagowaniem na potrzeby biznesu. Jednocześnie nasi inżynierowie rozumieją sens robienia konkretnych elementów z Backlogu i na Review mogą opowiedzieć po co go robili.” – **dodaje Maciej.**

F. Doskonałość techniczna

Do efektywnej współpracy wielu Drużyn, równie ważna jak efektywna komunikacja jest skuteczna synchronizacja na poziomie praktyk inżynierskich. W przypadku jednego, niewielkiego zespołu pracującego iteracyjnie długo „żyjące” osobne wersje kodu (tzw. branche) i późna integracja są wprawdzie groźne, ale dopiero w przypadku wielozespołowej pracy nad jednym produktem takie podejście zagwarantuje problemy. Dlatego środowisko pracy umożliwiające szybkie sprawdzenie, że wprowadzone zmiany działają zgodnie z założeniami i jednocześnie nie psują wcześniejszych funkcjonalności jest krytyczne. Oznacza to, że firmy pracujące w LeSS muszą wdrożyć praktyki inżynierskie pozwalające na szybkie uzyskanie informacji zwrotnej na temat jakości produktu. Łatwiej jednak powiedzieć, niż zrobić, a gdy pojawia się presja na osiągnięcie wyników biznesowych, to ta jakość może ucierpieć.

Niezawodność rozwiązania jest krytyczna w przypadku systemów telekomunikacyjnych, nad którymi pracuje Falcon V Systems. Z tego powodu nacisk na wdrożenie odpowiednich praktyk inżynierskich i dążenie do tzw. doskonałości technicznej (technical excellence) od początku były kluczowe. Na początku istniało jednak niewypowiedziane założenie, że to się „samo zadzieje” z inicjatywy różnych Drużyn. I trochę tak było, jednak często efekt nie był zadowalający. Chcąc upewnić się, że dostarczanie kolejnych funkcjonalności nie wpłynie negatywnie na jakość rozwiązania firma zdecydowała się wprowadzić rolę Technical Excellence Leada.

Skupił się on na stworzeniu środowiska pracy, które dawałoby inżynierom pewność, że wprowadzane zmiany spełnią standardy stawiane systemom telekomunikacyjnym.

„Przy pomocy Technical Excellence Leada usprawniliśmy znacząco systemy Ciągłej Integracji (Continuous Integration, CI), a także uściśliliśmy procesy automatyzacji testów i ich częstego wykonywania. Udało nam się wdrożyć statyczną analizę kodu i rozpocząć pracę nad systemami do Ciągłego Wdrożenia (Continuous Delivery, CD). Dzięki temu również nasi klienci mogą obecnie wykonać testy systemu na swoich środowiskach pracy.”

– **mówi Przemysław Górzyński, Software Developer w Falcon V Systems.**

Technical Excellence Lead stanowi więc formę piątego Scrum Mastera, tym razem skupionego wyłącznie na aspekcie inżynieryjnym.

G. Utrzymanie produktu

Kolejnym istotnym elementem pracy nad dużym produktem jest jego utrzymanie (maintenance). Tu również wiele firm tworzy osobne, dedykowane zespoły utrzymaniowe, ale jak wspomnieliśmy kilkakrotnie powyżej, grupy o wąskiej specjalizacji stanowią problem przy próbie zmiany priorytetów organizacji. Dodatkowo praca w takim zespole często jest kojarzona z powtarzającymi się, mało kreatywnymi zadaniami, więc raczej nie bywa popularna wśród inżynierów.

Z tego też powodu Falcon V Systems nie tworzył działu utrzymania, zakładając, że wszystkie Drużyny podzielą się tymi pracami po równo. Okazało się jednak, że jedna z nich brała na siebie coraz większą odpowiedzialność za te zadania. Chcąc uniknąć zlecenia prac utrzymaniowych tylko tej Drużynie po pewnym czasie w ramach Overall Retro wypracowano koncepcję „Duty Teamu”, co Sprint zmieniając Drużynę odpowiedzialną za te obowiązki. Gdy to nie wystarczało, żeby uporać się z narastającą listą tematów, firma eksperymentowała z dwoma Drużynami. Rezultaty wciąż jednak nie były zadowalające, dlatego w ramach kolejnych spotkań Overall Retro zespół analizował potencjalne przyczyny. Zauważył, że Duty Teamy w trakcie Sprintu poza utrzymaniem zajmują się również rozwojem produktu. Nudniejsze zadania często przegrywały z bardziej wymagającymi i utrzymanie nie dostawało wystarczającej uwagi. W konsekwencji ustalono, że Duty Team przez cały Sprint zajmuje się tylko i wyłącznie pracami związanymi z utrzymaniem. Dzięki temu firma unika konfliktu interesu i porzucania ważnych, ale mało ciekawych tematów na rzecz prac rozwojowych.

H. Podsumowanie

Ponad cztery lata wspólnej pracy nad produktem pozwoliło zespołowi Falcon V Systems zbudować silną i spójną kulturę organizacyjną. Od osób dołączających do firmy oczekuje on dużej odpowiedzialności, zarówno za tworzone rozwiązania, jak i za sam proces pracy. Zakłada, że gdy coś nie działa, to pracownicy będą o tym mówili i proponowali eksperymenty, które mogą zaadresować problem.

„Jednym z nieoczywistych objawów naszej kultury jest też specyficzne słownictwo stosowane w firmie - Ołweral (Overall Retro), Traveler, Rafinacja czy Drużyna, to unikalne dla nas rozwiązania, do których dochodziliśmy poprzez kolejne eksperymenty. I choć można je spotkać w innych firmach, to dla nas definiują bardzo konkretnie elementy naszego sposobu działania.”

– **zauważa Magdalena Dziegielewska, Scrum Master w Falcon V Systems**

Wewnątrz firmy udało się stworzyć silne, oparte na transparenacji fundamenty. Pozwalają one na otwartość, chęć współpracy, ale również powodują, że ludzie oczekują zaangażowania ich w decyzje i jasnych informacji co stoi za podejmowanymi wyborami.

Inwestycja w LeSSa nie da wartości z miesiąca na miesiąc, ale pozwoli na duży zwrot z inwestycji w dłuższym czasie. Zespół musiał się nauczyć współpracować razem zarówno wewnątrz Drużyn, jak i pomiędzy nimi. Zobaczył, jak można efektywnie propagować wiedzę i stworzyć adaptacyjną organizację mogącą szybko reagować na nowe informacje płynące z rynku.

Podziękowania

Chcielibyśmy bardzo podziękować wszystkim pracownikom Falcon V Systems zaangażowanym w powstanie tego dokumentu, w szczególności Małgorzacie Górskiej, COO, Bartoszowi Kajutowi, Członkowi Zarządu, Iwonie Leśnik, Marketing & Communication Manager, Przemysławowi Górzyńskiemu, Software Developer oraz Scrum Masterom: Karolinie Kędzierskiej, Magdalenie Dzięgielewskiej, Michałowi Jakubasowi i Maciejowi Słocińskiemu.

Jeżeli masz jakiegokolwiek pytania na temat Case Study to zapraszamy do kontaktu:

Tomasz Wykowski

Certified LeSS Coach, ProCognita

Tomasz.Wykowski@procognita.pl

<https://linkedin.com/in/wykowski>

Case Study powstało dzięki współpracy trwającej od 2019 roku oraz wywiadów przeprowadzonych w listopadzie 2022. Falcon V System wciąż się zmienia, jeżeli więc chcesz dowiedzieć się więcej na temat obecnej sytuacji napisz do Scrum Masterów firmy na adres sm@falconvsystems.com.

Możesz również skontaktować się z nimi na LinkedIn:

- Maciej Słociński <https://www.linkedin.com/in/maciejslocinski/>
- Michał Jakubas <https://www.linkedin.com/in/michal-jakubas/>
- Karolina Kędzierska <https://www.linkedin.com/in/karolinakedzierska1/>
- Magdalena Dzięgielewska <https://www.linkedin.com/in/magdadziegielewska/>