

Codewise Case Study

Agile w obsłudze klienta



O Codewise

Codewise¹ to firma, która z impetem wkroczyła na międzynarodowy rynek marketingu internetowego. Jako mały start-up podbili rynek, dostarczając nowatorskie i zaawansowane technologicznie produkty, które zrewolucjonizowały podejście do sprzedaży reklam w sieci. Odpowiadając na rosnące zapotrzebowanie oraz dynamikę rozwijającego się rynku firma z kilkuosobowego zespołu technicznych ekspertów w krótkim czasie urosła do ponad stu osób. Przy tak szybkim tempie wzrostu dotychczasowe metody pracy okazały się niewystarczające. Innymi słowy, wraz ze wzrostem firmy zwinność stała się wyzwaniem.



Cechą charakterystyczną Codewise, z której firma była dumna była silna kultura inżynierska oparta o samodzielność i kreatywność pracowników. I właśnie zachowanie tej kultury podczas zmian było dla organizacji priorytetem.

Jednym z zespołów, który został “przytłoczony sukcesem” firmy był Zespół Wsparcia Klienta (Customer Support). Portfolio dostarczanych produktów oraz liczba użytkowników rosły w ogromnym tempie, a to pociągało za sobą zwiększenie ilości i poziomu trudności zadań stojących przed tą grupą. Walcząc z natłokiem obowiązków zespół zdecydował się przetestować praktyki Scruma, aby odkryć efektywniejsze metody pracy.

Rozpoczynając eksperyment zespół borykał się z dwoma głównymi wyzwaniami:

- w związku z rosnącym zainteresowaniem produktami liczba zgłoszeń oraz klientów, których należało obsłużyć wzrosła w ostatnich miesiącach kilku- a w niektórych obszarach nawet kilkudziesięciokrotnie. Dlatego coraz trudniej było utrzymać

¹<https://codewise.com/>

wysoki poziom obsługi polegający na udzieleniu odpowiedzi klientowi w zakontraktowanym czasie (SLA - Service Level Agreement). Firma starała się zaadresować ten problem poprzez ustalenie optymalnej wielkości grupy, która byłaby w stanie obsłużyć wszystkie zgłoszenia, a jednocześnie wciąż efektywnie współpracować.

- przeciążenie pracą prowadziło do wypalenia zawodowego pracowników, a przez to do dużej rotacji w zespole. W ostatnich tygodniach przed rozpoczęciem eksperymentu zespół opuściło kilku doświadczonych członków. Napływ nowych pracowników tylko pogarszał sytuację, ponieważ czas wdrożenia nowej osoby, aby mogła wykonywać prace samodzielnie wynosił kilka tygodni i angażował najbardziej doświadczonych pracowników.

Etap 1: Rozpoznanie sytuacji, czyli sprawdzimy gdzie jesteśmy



Ilość niewiadomych była tak duża, że w pierwszej kolejności zespół postanowił poznać pole bitwy i skupił się na **wizualizacji prac**. Zauważył, że istnieją 2 podstawowe typy zadań:

- pytania klientów nazywane przez zespół **ticketami**. Setki zgłoszeń dziennie rejestrowanych automatycznie w systemie. Zespół nie miał żadnej kontroli nad tematyką, ilością zgłoszeń, ani czasem kiedy się pojawiają. Dodatkowo, aby zapewnić satysfakcję klientów, zgłoszenia miały ściśle wyznaczony czas na udzielenie odpowiedzi (SLA). Rodzaje pytań były różne od pomocy przy zakładaniu kont, migracji danych i wsparcia w pierwszych krokach używania systemu, po rozwiązywanie skomplikowanych problemów pojawiających się na styku rozwijanych systemów.

- zlecenia wewnętrzne, zwane przez zespół *zadaniami* - generowane przez inżynierów, dział marketingu, zarząd, a nawet sam zespół. Podobnie jak w przypadku ticketów zadania były bardzo różnorodne od pomocy przy weryfikacji płatności i przygotowywaniu danych po testowanie nowych funkcji. Tu nie obowiązywało SLA, ale większość z tych zadań była zazwyczaj potrzebna “natychmiast”.

Zespół zdecydował się użyć tablicy ściennej do zapisywania zadań, a tickety przechowywać nadal w narzędziu elektronicznym. Wybrał tygodniowy cykl pracy (**iteracji**), i już pierwszy tydzień przyniósł kilka interesujących spostrzeżeń.

- Po pierwsze, elementy na tablicy w ogóle się nie poruszały. Powody były różne: od zależności pomiędzy zadaniami, poprzez brak zrozumienia oczekiwań lub wiedzy niezbędnej do wykonania zadania, po zwyczajny brak czasu. Do pracy nad zadaniami członkowie zespołu zabierali się dopiero po uporaniu się z ticketami.
- Po drugie, zauważyli, że narzędzie elektroniczne, które rejestrowało pytania klientów nie dawało odpowiedzi na pytanie “jak szybko pracujemy?”, czyli ile ticketów zostało obsłużonych, czy zachowano właściwą kolejność, i czy spełniono wymagania SLA. Innymi słowy nie było żadnych danych do dalszej analizy.

W rezultacie zespół zdecydował się na dwie zmiany:

- drastyczne zmniejszenie liczby elementów na tablicy (zastosowanie zasady **Limit WIP**). Zamiast dotychczasowych kilkadziesiąt zadań zespół wybierał 1-2 zadania, które realizował w czasie iteracji. Pozwoliło to skupić się na nich, a dzięki temu dostarczać je szybciej i bardziej przewidywalnie. Pozostałym zgłoszeniom zespół mówił „NIE TERAZ”. Część zgłaszających doceniła szczerą odpowiedź, inni przekazywali swoją frustrację, ale nie była ona większa niż poprzednio, gdy dowiadawali się, że ich zadanie, które zespół przyjął nie zostało zrealizowane.
- zidentyfikowanie i wprowadzenie narzędzia, które pomogłoby analizować trendy i wzorce w ticketach. Ponieważ nowe narzędzie było łatwiejsze w użyciu, zespół zaczął szybciej obsługiwać zgłoszenia klientów. Dodatkowo zebrane dane pozwoliły ustalić z jakimi pytaniami klienci zgłaszali się najczęściej. Zarówno te informacje, jak i uwolniony czas stały się kluczowe podczas dalszych prac.

Rezultatem kilku tygodni pracy w tym modelu była świadomość przepustowości zespołu, oraz wytypowanie obszarów wymagających najwięcej uwagi.

Etap 2: Ewolucja zespołu

Zespół w tym momencie składał się z 2 doświadczonych i 6 nowo zatrudnionych osób. Większość pracy polegała na odpowiadaniu na przydzielone przez system tickety. Ponieważ praca była wykonywana indywidualnie przez członków zespołu proces dzielenia się wiedzą był ograniczony do sesji wprowadzających dla nowych osób. Praca była też często powielana, gdy kilka osób równocześnie szukało rozwiązania podobnego problemu zgłoszonego przez różnych klientów. Nowi członkowie zespołu nie mogli realizować obowiązków z powodu braku wiedzy i informacji, podczas gdy doświadczone osoby były przeładowane pracą. Aby poprawić wymianę informacji między członkami zespołu, wprowadził codzienne spotkania synchronizujące (**Daily Scrum**). Pierwsze spotkania trwały dość długo, ponieważ służyły zespołowi nie tylko do koordynacji prac i omawiania przeszkód, ale również do rozwiązywania trudniejszych ticketów i zadań. Dość szybko zespół zauważył, że do rozwiązania takich problemów wystarczały zazwyczaj dwie osoby. Bazując na tej obserwacji zaadaptował na swoje potrzeby popularną wśród inżynierów technikę **“pair programming”**, polegającą na wykonywaniu prac przez dwie osoby i nazwał ją **“shadowing”**. Podczas spotkania zespół ustalał przy jakich zadaniach shadowing jest potrzebny i jaki powinien być skład par w danym dniu.

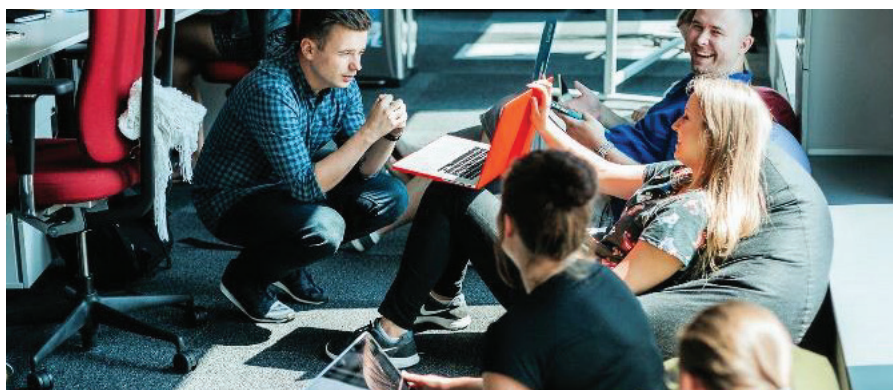
Dzięki temu wytworzył się nawyk współpracy pomiędzy członkami zespołu w trakcie dnia, a czas spotkania synchronizującego spadł poniżej 15 minut. Znacząco skrócił się też czas wdrażania nowozatrudnionych, którzy już od pierwszego dnia pracowali w systemie **“shadowingu”**. Dodatkową zaletą było zacieśnienie relacji towarzyskich w zespole i poprawa moralności. Pracownicy zaczęli się mocno identyfikować ze swoim zespołem.

Etap 3: Ewolucja zespołu

Nie ma możliwości przewidzenia ani kontrolowania liczby zgłoszeń klientów, co w dużym stopniu wpływa na proces planowania prac w dziale obsługi klienta. Zapytania różnią się od siebie i odpowiedź na niektóre z nich zajmuje chwilę, podczas gdy inne wymagają zastanowienia, zgromadzenia danych lub dokładnej analizy. Oprogramowanie wysyłało informację o każdym nowym tickecie do członków zespołu niezależnie czy skończyli już pracę nad wcześniejszym zgłoszeniem czy jeszcze nie. Zadania zgłaszane przez działy wewnętrzne miały nieco inną specyfikę i zazwyczaj były dość pracołłonne. Ponieważ pytania klientów mają dla

zespołu wsparcia zawsze najwyższy priorytet, każdy nowy ticket odrywał pracownika od zadania, nad którym pracował, co prowadziło do ciągłego przełączanie kontekstu i miało wpływ na efektywność członków zespołu. Aby temu zaradzić, zespół wprowadził zasadę pełnego skupienia, którą nazwał “No ticket day”. Zgodnie z wcześniejszymi ustaleniami podczas planowania iteracji zespół wybierał kilka zadań, które chciał zrealizować. Podczas codziennego spotkania członkowie zespołu ustalali, kto w danym dniu będzie pracował nad większym zadaniem wymagającym skupienia. Aby pomóc takiej osobie reszta zespołu chroniła ją przed zakłóceniami, przekierowując na siebie większą ilość pytań klientów.

Ta zmiana sprawiła, że czas realizacji (**cycle time**) zadań i bardziej skomplikowanych ticketów znacząco się zmniejszył.



Etap 4: Automatyzacja

Każda iteracja kończyła się spotkaniem będącym odpowiednikiem Review i Retrospektywy w Scrumie. Zespół analizował dane z nowego narzędzia do zarządzania zgłoszeniami dotyczące typów zgłoszeń klientów, informacje o awariach oraz nowe funkcje dostarczane przez programistów. Podczas tych spotkań zespół identyfikował najbardziej czasochłonne, najczęstsze oraz najbardziej zniechęcające typy ticketów i zadań, a następnie stopniowo redukował ich ilość. Korzystając z zasady skupienia oraz z czasu zaoszczędzonego dzięki wcześniejszym zmianom część pracy zespołu została **zautomatyzowana**. Stworzenie przez zespół samouczków, filmów instruktażowych i bazy wiedzy dla użytkowników oraz repozytorium gotowych odpowiedzi na najczęstsze pytania dało przestrzeń do nowych eksperymentów i uwolniło członków zespołu od części powtarzalnej pracy.

Startując od zajmowania się wyłącznie odpowiadaniem na zgłoszenia klientów zespół osiągnął równowagę wynoszącą około 60% czasu spędzonego pracy nad ticketami i 40%, które poświęcał na ulepszenia oraz obsługę zadań z wnętrza organizacji.

Efekt: Zgrany zespół obsługi klienta

Po roku od rozpoczęcia eksperymentów zespół powiększył się i podzielił na grupy bardziej skupione wokół produktów. Wszyscy nadal używają technik Scrumowych pracując w tygodniowych iteracjach, zapewniając wizualizację zadań przy pomocy tablic i koordynując pracę poprzez Daily Scrum. Praktyka pracy w parach (“shadowing”) i polityka skupienia (“no ticket day”) rozprzestrzeniły się na większość innych działów w organizacji. Retrospektywa i Planowanie Sprintu zostały połączone w jedno spotkanie, ponieważ plany zespołu zaczęły się koncentrować głównie na ulepszeniach modelu pracy. Zespół zdecydował, że nie tylko spełnia wymogi SLA, ale jest gotowy je rozszerzyć do wsparcia 24/7. Przygotował program wdrożeniowy używany nie tylko przez nowych członków, ale także przez inne zespoły i klientów. Nowi pracownicy są gotowi do podjęcia samodzielnej pracy w krótszym czasie niż przed transformacją. Rotacja w zespole spadła niemal do zera. W ciągu ostatniego roku nikt nie opuścił zespołu, a kiedy członkowie zespołu mieli możliwość zmiany działu, prawie wszyscy zdecydowali się pozostać w dziale obsługi klienta.

Eksperymenty działu wsparcia zaowocowały rozszerzeniem użycia praktyk Scrum w całej organizacji. Część zespołów IT zaczęła przeprowadzać Review wspólnie z działem obsługi klienta. Inne zapraszają go na swoje planowania i angażują w codzienną pracę grupy programistycznej. Firma wciąż się rozwija i wprowadza nowe produkty.

Podziękowanie

Składamy serdeczne podziękowania wszystkim członkom zespołu i menedżerom, którzy z nami współpracowali, za cierpliwość i dociekliwość, za pogodzenie się ze slangiem IT takim jak “refactoring”, „automatyzacja”, „pair programming”, oraz za kreatywność w przenoszeniu wzorców ze świata oprogramowania na nowy grunt.

Szczególnie dziękujemy Tomaszowi Kaczanowskiemu, Aleksandrowi Fronczek i Maciejowi Szlachcie za zaproszenie i wspieranie naszych eksperymentów.

Jeżeli chcesz wiedzieć więcej zapraszam do kontaktu,

Justyna Wykowska,

ProCognita,

Justyna.Wykowska@procognita.pl