

Opracowanie podsystemów sprawdzających zadania dla różnych języków dla Systemu STOS

Kanban: Tablica Kanban

Wersja 1.0.0

Data utworzenia obecnej wersji: 13-05-2025

Data utworzenia dokumentu: 13-04-2025

Mateusz Fydrych
193410

Damian Trowski
193443

Maciej Raciniewski
189774

Spis treści

1	O projekcie i produkcji	2
2	Stany zgłoszeń/zadań	2
3	Limity WIP (Work In Progress)	4
4	Tablica Kanban	4
5	Metryki produktywności	5

1 O projekcie i produkcji

Projekt dotyczy opracowania podsystemów sprawdzających zadania dla różnych języków programowania dla Systemu STOS. System ten będzie służył do automatycznego sprawdzania zadań programistycznych, co usprawni proces oceniania prac studentów oraz zapewni jednolite kryteria sprawdzania. Poszczególne podsystemy zostaną zrealizowane w formie oddzielnych kontenerów, aby wykorzystać modularność głównego systemu i umożliwić efektywne skalowanie całego rozwiązania.

W ramach oceny zadania pierwszym krokiem jest kompilacja kodu źródłowego, za którą odpowiada kontener do kompilacji danego języka. Jeżeli proces przebiegnie poprawnie (kod źródłowy jest poprawny), skompilowany program trafia do kontenera odpowiadającego za uruchamianie go dla każdego przypadku testowego podanego na wejściu (pliki w postaci 0-n.in). Następnie kontener zapisuje wyniki działania dla każdego przypadku w osobnych plikach (0-n.stdout.out i 0-n.stderr.out) oraz dane takie jak czas działania i zużycie zasobów. Ostatnim etapem jest kontener oceniający, który porównuje wyjście programu z poprawnymi odpowiedziami i dla każdego testu wystawia ocenę.

2 Stany zgłoszeń/zadań

W tablicy Kanban dla projektu zdefiniowano następujące stany (kolumny), przez które przechodzić będzie każde zgłoszenie/zadanie. Zestaw stanów został dostosowany do specyfiki pracy zespołu i projektu.

TO DO (Do zrobienia)

Opis: Zadania zaplanowane do realizacji w bieżącym sprincie lub oczekujące na podjęcie. Znajdują się tu elementy, które zostały zidentyfikowane jako praca do wykonania, ale jeszcze nie rozpoczęto nad nimi aktywnych działań.

Kryteria przejścia do następnych stanów:

- → **IN PROGRESS:** Rozpoczęcie aktywnej pracy nad zadaniem przez członka zespołu.
- (Rzadziej) → **ON HOLD:** Zidentyfikowano przeszkodę uniemożliwiającą rozpoczęcie zadania (np. brak zasobów, niejasne wymagania).

IN PROGRESS (W toku)

Opis: Zadania, nad którymi aktualnie trwają prace deweloperskie (implementacja, kodowanie, tworzenie).

Kryteria przejścia do następnych stanów:

- → **TESTING:** Ukończono implementację podstawowej funkcjonalności i zadanie jest gotowe do przeprowadzenia testów.
- → **ON HOLD:** Zidentyfikowano przeszkodę (np. brak informacji, zależność od innego zadania, problem techniczny), która uniemożliwia dalszą pracę.
- (Rzadziej) → **TO DO:** Zadanie zostało cofnięte do puli (np. z powodu zmiany priorytetów lub konieczności fundamentalnej zmiany zakresu).

ON HOLD (Wstrzymane)

Opis: Zadania, których realizacja jest tymczasowo wstrzymana z powodu zewnętrznych lub wewnętrznych przeszkód, zależności, oczekiwania na informacje, decyzje, lub inne czynniki, które uniemożliwiają natychmiastowy dalszy postęp.

Kryteria przejścia do następnych stanów:

- → **TO DO:** Przeszkoda została usunięta, ale zadanie nie było jeszcze w zaawansowanym etapie realizacji lub wymaga ponownego zaplanowania; wraca do puli zadań do wykonania.
- → **IN PROGRESS:** Przeszkoda/powód wstrzymania został usunięty, a zadanie było wcześniej w trakcie implementacji lub może teraz rozpocząć/kontynuować implementację.

- → **TESTING:** Przeszkoda/powód wstrzymania został usunięty, a zadanie było wcześniej wstrzymane podczas etapu testowania i może teraz kontynuować testy.
- → **REVIEW:** Przeszkoda/powód wstrzymania został usunięty, a zadanie było wcześniej wstrzymane podczas etapu przeglądu i może teraz kontynuować przegląd.

TESTING (W trakcie testowania)

Opis: Zadania poddawane procesowi weryfikacji (np. testy jednostkowe, integracyjne, użytkownika) w celu sprawdzenia poprawności działania zaimplementowanej funkcjonalności i zgodności z wymaganiami.

Kryteria przejścia do następnych stanów:

- → **REVIEW:** Testy zakończyły się pomyślnie, zadanie jest gotowe do przeglądu kodu lub finalnej akceptacji.
- → **IN PROGRESS:** Testy wykazały błędy lub niezgodności, zadanie wymaga poprawek i wraca do etapu deweloperskiego.
- → **ON HOLD:** Zidentyfikowano przeszkodę uniemożliwiającą kontynuację testów (np. problemy ze środowiskiem testowym, brak danych testowych).

REVIEW (W przeglądzie)

Opis: Kod zadania, jego implementacja lub wyniki są przeglądane przez co najmniej jednego innego członka zespołu (code review) w celu zapewnienia jakości, zgodności ze standardami i wykrycia potencjalnych problemów przed finalnym zatwierdzeniem.

Kryteria przejścia do następnych stanów:

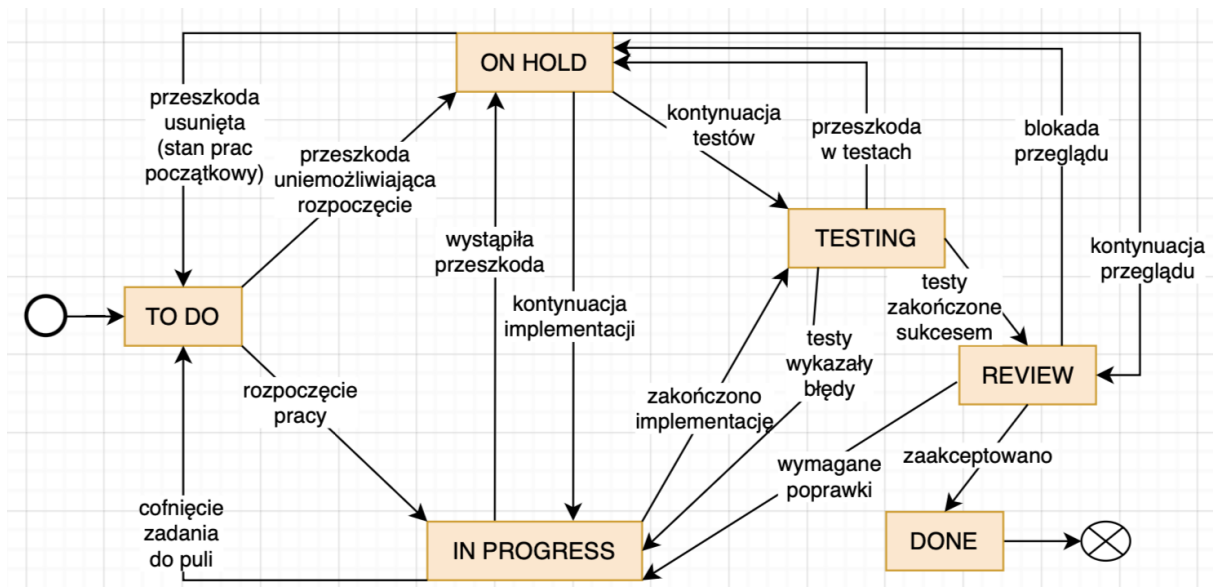
- → **DONE:** Zadanie zostało pomyślnie zweryfikowane podczas przeglądu, zaakceptowane i spełnia wszystkie kryteria ukończenia.
- → **IN PROGRESS:** Przegląd wykazał konieczność wprowadzenia zmian lub poprawek w kodzie lub implementacji; zadanie wraca do etapu deweloperskiego.
- → **ON HOLD:** Zidentyfikowano przeszkodę uniemożliwiającą zakończenie przeglądu (np. brak dostępności recenzenta z odpowiednią wiedzą, oczekiwanie na decyzję zewnętrzną).

DONE (Ukończone)

Opis: Zadania, które zostały pomyślnie zaimplementowane, przetestowane, zrecenzowane i spełniają wszystkie kryteria z Definicji Ukończenia (Definition of Done). Praca nad tymi zadaniami została zakończona.

Kryterium przejścia:

Jest to stan końcowy w przepływie pracy dla danego zadania.



Rysunek 1: Diagram możliwych przejść między stanami

3 Limity WIP (Work In Progress)

W projekcie z trzyosobowym zespołem ustalono limity WIP (maksymalna liczba zadań w danej kolumnie Kanban) dla każdej aktywnej kolumny (**IN PROGRESS**, **ON HOLD**, **TESTING**, **REVIEW**) na poziomie 3 zadań. Wynika to z liczby deweloperów – jeśli w danym momencie w którejś kolumnie znajdowałoby się więcej niż 3 zadania, oznacza to, że występuje zator i należy wpierw nadrobić zaległości w innych etapach pracy.

Limity te mają na celu:

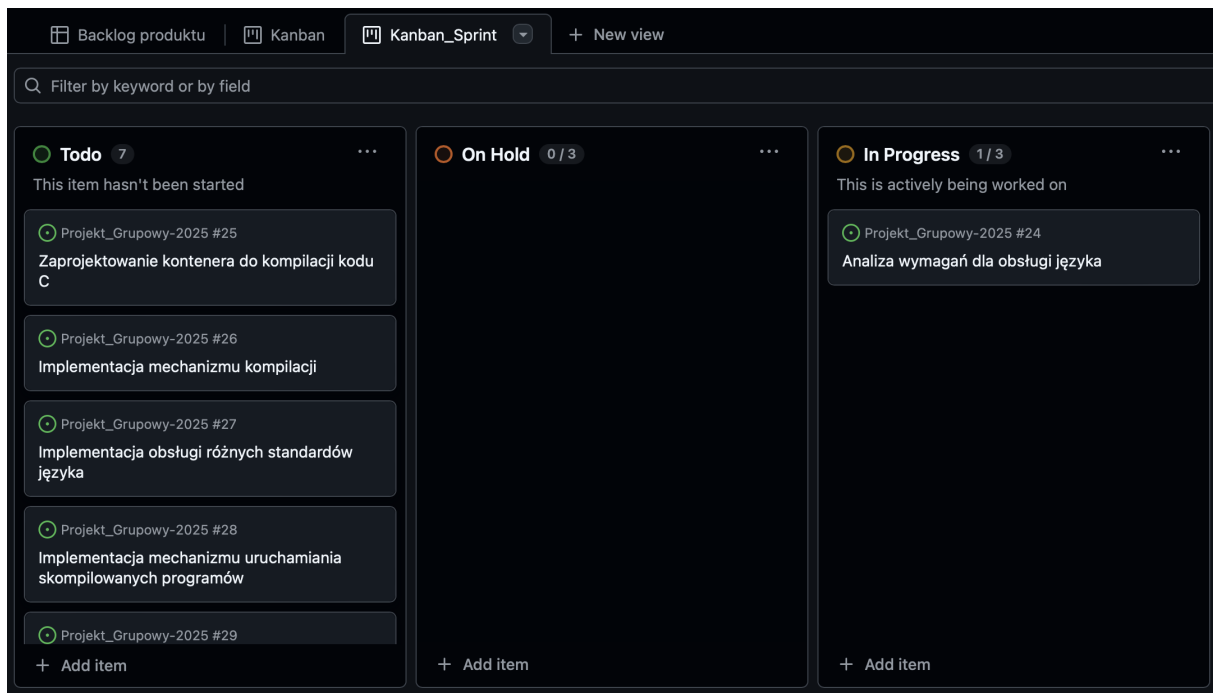
- unikanie przeciążenia zespołu,
- minimalizowanie multitaskingu,
- zapobieganie tworzeniu wąskich gardeł,
- utrzymanie płynnego przepływu pracy.

Kolumny **TO DO** i **DONE** nie mają limitów.

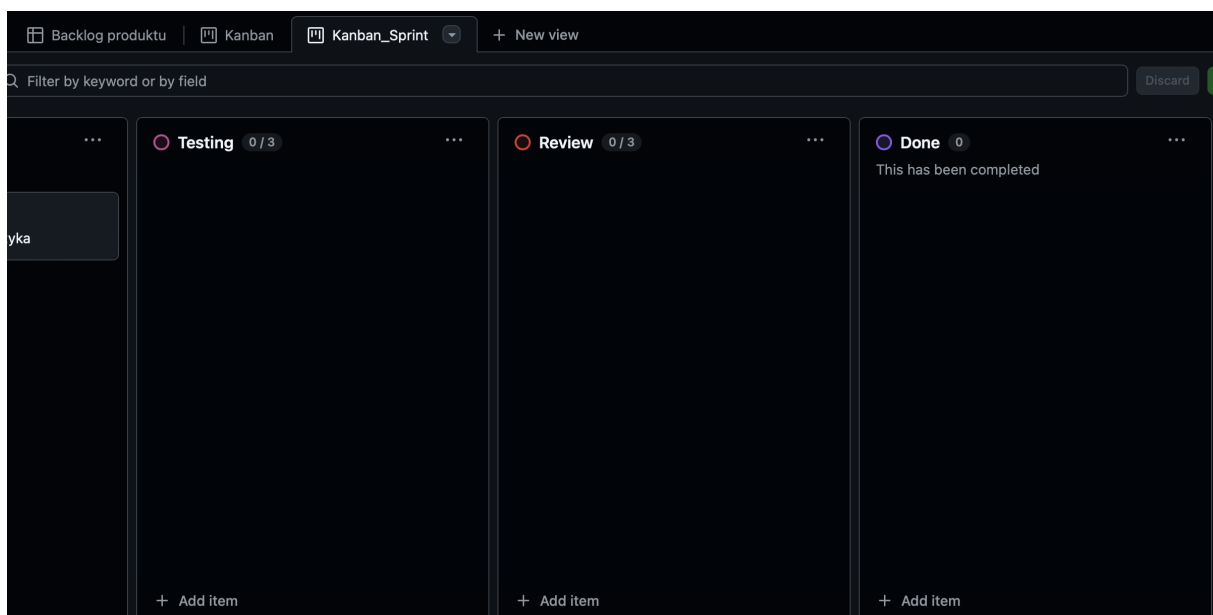
4 Tablica Kanban

Tablica Kanban dla bieżącego sprintu jest zaimplementowana w narzędziu wspomagającym GitHub Projects, które jest zintegrowane z repozytorium kodu. Tablica ta wizualizuje przepływ pracy, zestawiając elementy backlogu produktu wybrane do realizacji w sprincie.

Karty na tablicy reprezentują poszczególne zadania i są umieszczone w kolumnach odpowiadających ich aktualnemu stanowi realizacji: **TO DO**, **IN PROGRESS**, **ON HOLD**, **TESTING**, **REVIEW**, **DONE**.



Rysunek 2: Tablica Kanban cz.1



Rysunek 3: Tablica Kanban cz.2

5 Metryki produktywności

W celu monitorowania wydajności przepływu pracy, identyfikacji obszarów do usprawnienia oraz oceny produktywności zespołu, zdefiniowano następujące metryki:

Czas cyklu (Cycle Time)

Opis: Mierzy czas od momentu, gdy zadanie faktycznie rozpoczyna być przetwarzane (przejście ze stanu **TO DO** do **IN PROGRESS**) do momentu jego ukończenia (przejście do stanu **DONE**). Obejmuje czas spędzony w stanach **IN PROGRESS**, **ON HOLD**, **TESTING**, **REVIEW**.

Cel: Pomaga zrozumieć, ile czasu zespół potrzebuje na realizację zadania od rozpoczęcia aktywnej pracy. Skracanie czasu cyklu jest często celem zespołów stosujących Kanban.

Czas przebywania w stanie

Opis: Mierzy czas, jaki każde zadanie spędza w danym stanie (np. **IN PROGRESS**, **TESTING**, **REVIEW**, **ON HOLD**). Pozwala analizować, które etapy procesu są najbardziej czasochłonne lub narażone na przestoje.

Cel: Ułatwia identyfikację wąskich gardeł oraz usprawnianie konkretnych etapów przepływu pracy poprzez skracanie czasu spędzanego w poszczególnych kolumnach.

Czas realizacji (Lead Time)

Opis: Mierzy całkowity czas od momentu zgłoszenia zadania lub dodania go do backlogu sprintu (np. pojawienie się w kolumnie **TO DO**) do jego faktycznego ukończenia (przejście do stanu **DONE**). Obejmuje cały czas oczekiwania i przetwarzania.

Cel: Daje perspektywę na to, jak długo trwa dostarczenie wartości od momentu jej zidentyfikowania/zaplanowania.

Liczba zadań w toku (Work In Progress - WIP)

Opis: Bieżąca liczba kart (zadań) znajdujących się we wszystkich aktywnych stanach przepływu (tj. **IN PROGRESS**, **ON HOLD**, **TESTING**, **REVIEW**).

Cel: Monitorowanie zgodności z ustalonymi limitami WIP, identyfikacja wąskich gardeł (gdzie zadania kumulują się nadmiernie) oraz zapewnienie płynności przepływu.

Przepustowość (Throughput)

Opis: Liczba zadań ukończonych (przeniesionych do stanu **DONE**) w określonym przedziale czasu (np. na tydzień, na sprint).

Cel: Wskazuje na tempo dostarczania wartości przez zespół. Może być używana do prognozowania przyszłej wydajności.

Wybrane metryki będą śledzone i analizowane, z wykorzystaniem dostępnych narzędzi. Prezentacja i omówienie tych metryk będą omówione na spotkaniu „retrospektywa sprintu”.