



# PROJEKT INŻYNIERSKI

## Edytor graficzny systemów rozmytych dla języka Python

ID projektu - 46300

Opiekun projektu - dr inż. Jerzy Dembski

### Dokument nr 4: Kanban: Tablica Kanban

#### Streszczenie projektu:

Celem projektu jest tworzenie edytora graficznego systemu rozmytego z wykorzystaniem dowolnej biblioteki Pythona (np. pygame, opencv, opengl) pozwalającego na tworzenie i kształtowanie zbiorów rozmytych, definiowanie reguł rozmytych, wizualizację działania systemu dla zadanych wartości wejściowych oraz uczenie systemu na podstawie danych uczących metodą ANFIS wraz z przedstawieniem systemu w postaci wielowarstwowego modelu neuronowego do dalszego uczenia. System powinien pozwalać też na zapis i odczyt systemu rozmytego z pliku tekstowego, jak również przedstawienia go jako funkcji przetwarzającej dane wejściowe.

#### Streszczenie dokumentu:

Celem dokumentu jest opracowanie tablicy Kanban dla aktualnego sprintu.

Wersja:	1.2
Data wydania:	29.04.2025
Redaktor:	Adam Zarzycki
Współautorzy:	Filip Wesołowski, Julian Kulikowski
Etap/zadanie:	4
Nazwa pliku:	Kanban_v1.2.pdf
Liczba stron:	20



## Historia zmian

Wersja	Data	Opis zmiany
1.0	25.04.2025	Dodanie punktów 1. i 2.
1.1	28.04.2025	Dodanie punktów 3. i 4.
1.2	29.04.2025	Dodanie punktu 5.





## SPIS TREŚCI

<b>1. O projekcie i produkcji</b>	<b>5</b>
<b>2. Stany zgłoszeń / zadań</b>	<b>6</b>
2.1. TO_DO	6
2.1.1. Definicja	6
2.1.2. Stany wejściowe dla stanu	6
2.1.3. Stany wyjściowe dla stanu	6
2.1.4. Kryteria wyjścia ze stanu	6
2.2. IN_PROGRESS	6
2.2.1. Definicja	6
2.2.2. Stany wejściowe dla stanu	6
2.2.3. Stany wyjściowe dla stanu	6
2.2.4. Kryteria wyjścia ze stanu	7
2.3. ON_HOLD	7
2.3.1. Definicja	7
2.3.2. Stany wejściowe dla stanu	7
2.3.3. Stany wyjściowe dla stanu	7
2.3.4. Kryteria wyjścia ze stanu	7
2.4. IN_TEST	7
2.4.1. Definicja	7
2.4.2. Stany wejściowe dla stanu	8
2.4.3. Stany wyjściowe dla stanu	8
2.4.4. Kryteria wyjścia ze stanu	8
2.5. IN_PEER_REVIEW	8
2.5.1. Definicja	8
2.5.2. Stany wejściowe dla stanu	8
2.5.3. Stany wyjściowe dla stanu	8
2.5.4. Kryteria wyjścia ze stanu	8
2.6. IN_SUPERVISOR_REVIEW	8
2.6.1. Definicja	9
2.6.2. Stany wejściowe dla stanu	9
2.6.3. Stany wyjściowe dla stanu	9
2.6.4. Kryteria wyjścia ze stanu	9
2.7. DONE	9
2.7.1. Definicja	9
2.7.2. Stany wejściowe dla stanu	9
2.7.3. Stany wyjściowe dla stanu	9
2.7.4. Kryteria wyjścia ze stanu	9
2.8. BLOCKED	9
2.8.1. Definicja	9
2.8.2. Stany wejściowe dla stanu	10
2.8.3. Stany wyjściowe dla stanu	10
2.8.4. Kryteria wyjścia ze stanu	10



2.9. CANCELLED	10
2.9.1. Definicja	10
2.9.2. Stany wejściowe dla stanu	10
2.9.3. Stany wyjściowe dla stanu	10
2.9.4. Kryteria wyjścia ze stanu	10
2.10. Diagram przejść pomiędzy stanami	10
2.10.1. Założenia	11
2.10.2. Diagram przejść	11
<b>3. Limity WIP</b>	<b>12</b>
3.1. TO_DO	12
3.2. IN_PROGRESS	12
3.3. ON_HOLD	12
3.4. IN_TEST	12
3.5. IN_PEER_REVIEW	12
3.6. IN_SUPERVISOR_REVIEW	12
3.7. DONE	13
3.8. BLOCKED	13
3.9. CANCELLED	13
<b>4. Tablica Kanban</b>	<b>14</b>
<b>5. Metryki produktywności</b>	<b>19</b>
5.1. Przepustowość	19
5.1.1. Definicja	19
5.1.2. Cel	19
5.2. Lead Time	19
5.2.1. Definicja	19
5.2.2. Cel	19
5.3. Długość cyklu	19
5.3.1. Definicja	19
5.3.2. Cel	19
5.4. Praca w toku (WIP)	20
5.4.1. Definicja	20
5.4.2. Cel	20
5.5. Czas przebywania zadania w określonym stanie	20
5.5.1. Definicja	20
5.5.2. Cel	20

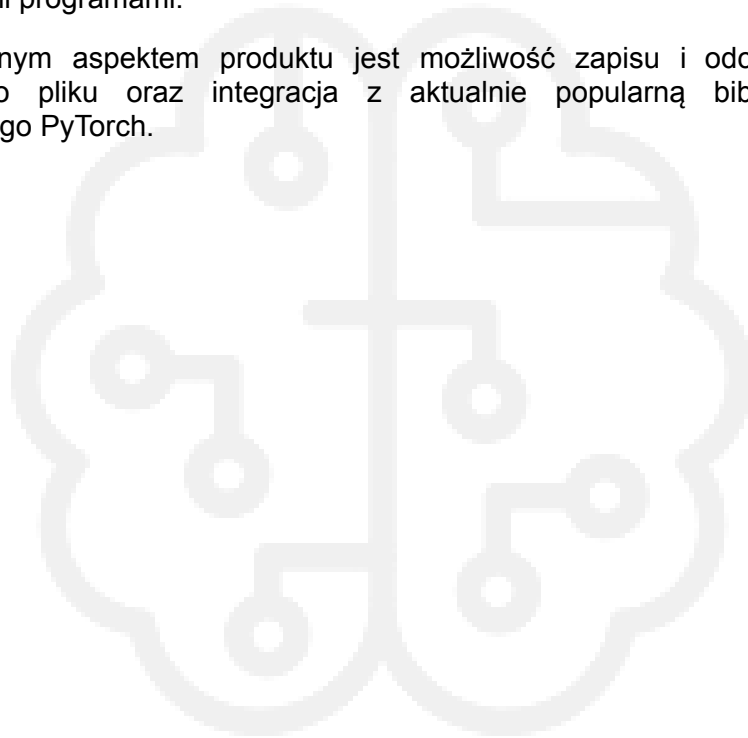


# 1. O projekcie i produkcie

Celem końcowym projektu jest utworzenie aplikacji umożliwiającej wizualizację zbiorów rozmytych za pomocą interfejsu graficznego w języku Python. Ważnym aspektem produktu jest udostępnienie darmowej alternatywy dla programu Fuzzy Logic Designer operującego w płatnym środowisku MATLAB.

Najważniejszym zadaniem produktu jest stworzenie prostego, przystępnego interfejsu graficznego, umożliwiającego użycie aktualnych implementacji logiki rozmytej. Użytkownik powinien być w stanie określać oraz edytować reguły rozmyte, a następnie obserwować ich efekt oraz podejmowanie decyzji w czasie rzeczywistym. WŚ przeciwieństwie do aktualnych rozwiązań korzystających z wiersza poleceń, naszym priorytetem jest obserwowalność wyników, kontrolowalność kolejnych kroków oraz przystępność dla niedoświadczonych użytkowników. Powinien ich naprowadzać oraz uczyć w naturalny sposób, nawet jeżeli nigdy wcześniej nie pracowali z podobnymi programami.

Innym ważnym aspektem produktu jest możliwość zapisu i odczytu aktualnego projektu do pliku oraz integracja z aktualnie popularną biblioteką uczenia maszynowego PyTorch.





## 2. Stany zgłoszeń / zadań

### 2.1. TO\_DO

#### 2.1.1. Definicja

Jeden z dwóch stanów początkowych (razem z BLOCKED). Oznacza zadanie, które należy wykonać w bieżącym sprincie, jednak praca nad nim jeszcze się nie rozpoczęła, ani nie został do niego przypisany żaden członek zespołu projektowego.

#### 2.1.2. Stany wejściowe dla stanu

Brak - stan początkowy.

#### 2.1.3. Stany wyjściowe dla stanu

- IN\_PROGRESS,
- BLOCKED,
- CANCELLED.

#### 2.1.4. Kryteria wyjścia ze stanu

- gdy członek zespołu projektowego zostanie przypisany do zadania i podejmie jego realizację, zadanie przechodzi do stanu IN\_PROGRESS,
- gdy dodane zostanie inne zadanie, które musi zostać wykonane, by realizacja tego zadania była możliwa, przechodzi ono do stanu BLOCKED,
- gdy wykonanie zadania w obecnym sprincie zostaje porzucone lub interesariusze rezygnują z implementacji w ostatecznym produkcie funkcjonalności realizowanej w ramach zadania, przechodzi ono do stanu CANCELLED.

### 2.2. IN\_PROGRESS

#### 2.2.1. Definicja

Zadanie jest aktywnie wykonywane przez przypisanego do niego członka zespołu projektowego.

#### 2.2.2. Stany wejściowe dla stanu

- TO\_DO,
- ON\_HOLD.

#### 2.2.3. Stany wyjściowe dla stanu

- ON\_HOLD,
- IN\_TEST.



#### **2.2.4. Kryteria wyjścia ze stanu**

- gdy praca nad zadaniem zostaje przerwana, a członek zespołu projektowego zostaje przypisany do innego zadania, przechodzi ono do stanu ON\_HOLD (dotyczy również sytuacji blokowania lub anulowania zadań - najpierw zwalnia się z nich członków zespołu projektowego),
- gdy zadanie zostaje wykonane, członek zespołu projektowego przekazuje je testerowi do oceny poprawności wykonania i zaczyna realizować kolejne zadanie - przechodzi ono do stanu IN\_TEST.

### **2.3. ON\_HOLD**

#### **2.3.1. Definicja**

Zadanie zostało wykonane częściowo, ale w chwili obecnej nie jest wykonywane i żaden członek zespołu projektowego nie jest do niego przypisany.

#### **2.3.2. Stany wejściowe dla stanu**

- IN\_PROGRESS,
- IN\_TEST,
- IN\_PEER\_REVIEW,
- IN\_SUPERVISOR\_REVIEW,
- BLOCKED.

#### **2.3.3. Stany wyjściowe dla stanu**

- IN\_PROGRESS,
- BLOCKED,
- CANCELLED.

#### **2.3.4. Kryteria wyjścia ze stanu**

- gdy jakiś członek zespołu projektowego podejmuje ponownie pracę nad danym zadaniem, przechodzi ono do stanu IN\_PROGRESS,
- gdy dodane zostaje kolejne zadanie, które musi być wykonane przed kontynuacją tego zadania, przechodzi ono do stanu BLOCKED,
- gdy pełne wykonanie zadania w obecnym sprincie zostaje porzucone lub interesariusze rezygnują z implementacji w ostatecznym produkcie funkcjonalności zrealizowanej częściowo w ramach tego zadania, przechodzi ono do stanu CANCELLED.

### **2.4. IN\_TEST**

#### **2.4.1. Definicja**



Zadanie jest testowane przez przypisanego testera (członka zespołu projektowego), w celu sprawdzenia poprawności implementacji i wykrycia potencjalnych błędów.

### **2.4.2. Stany wejściowe dla stanu**

- IN\_PROGRESS.

### **2.4.3. Stany wyjściowe dla stanu**

- ON\_HOLD,
- IN\_PEER\_REVIEW.

### **2.4.4. Kryteria wyjścia ze stanu**

- gdy znalezione zostaną błędy implementacji i/lub błędy projektowe wykonania zadania, które wymagają poprawek, wraca ono do etapu implementacji - przechodzi do stanu ON\_HOLD,
- gdy nie zostaną znalezione żadne błędy lub poprawa tychże błędów zostanie uznana za nieopłacalną zgodnie z przyjętą metodyką, zadanie zostaje przekazane innemu członkowi zespołu projektowego do oceny - przechodzi do stanu IN\_PEER\_REVIEW.

## **2.5. IN\_PEER\_REVIEW**

### **2.5.1. Definicja**

Formalna poprawność wykonania zadania i jego zgodność z wymaganiami sprawdzana jest przez innego, niezaangażowanego bezpośrednio w jego wykonanie (dla zachowania bezstronności opinii i innego spojrzenia na problem) członka zespołu projektowego.

### **2.5.2. Stany wejściowe dla stanu**

- IN\_TEST.

### **2.5.3. Stany wyjściowe dla stanu**

- ON\_HOLD,
- IN\_SUPERVISOR\_REVIEW.

### **2.5.4. Kryteria wyjścia ze stanu**

- gdy w wykonaniu zadania znalezione zostaną błędy wykonania zadania lub w jakikolwiek inny sposób wymaga ono poprawek, przechodzi ono do stanu ON\_HOLD (wymaga ponownego podjęcia przez któregoś członka zespołu projektowego),
- gdy oceniający nie będzie miał zastrzeżeń do wykonanego zadania, przekazuje je opiekunowi projektu do oceny - przechodzi ono do stanu IN\_SUPERVISOR\_REVIEW.

## **2.6. IN\_SUPERVISOR\_REVIEW**





### 2.6.1. Definicja

Zadanie jest oceniane przez opiekuna projektu (dra inż. Jerzego Dembskiego) pod względem zgodności z określonymi na początku wymaganiami.

### 2.6.2. Stany wejściowe dla stanu

- IN\_PEER\_REVIEW.

### 2.6.3. Stany wyjściowe dla stanu

- ON\_HOLD,
- DONE.

### 2.6.4. Kryteria wyjścia ze stanu

- gdy w wykonaniu zadania znalezione zostaną niezgodności z wymaganiami opiekuna projektu lub w jakikolwiek inny sposób wymaga ono poprawek, przechodzi ono do stanu ON\_HOLD (wymaga ponownego podjęcia przez któregoś członka zespołu projektowego),
- gdy opiekun nie będzie miał zastrzeżeń do wykonanego zadania, zostaje ono uznane za wykonane - przechodzi do stanu DONE.

## 2.7. DONE

### 2.7.1. Definicja

Jeden z dwóch stanów końcowych (razem z CANCELLED). Zadanie zostało wykonane, przetestowane, i sprawdzone pod względem wymagań określonych przez opiekuna projektu. Praca nad nim zakończyła się sukcesem.

### 2.7.2. Stany wejściowe dla stanu

- IN\_SUPERVISOR\_REVIEW.

### 2.7.3. Stany wyjściowe dla stanu

Brak - stan końcowy.

### 2.7.4. Kryteria wyjścia ze stanu

Brak - stan końcowy.

## 2.8. BLOCKED

### 2.8.1. Definicja

Jeden z dwóch stanów początkowych (razem z TO\_DO). Zadanie jest zablokowane - wymagane jest wcześniejsze zrealizowanie innego zadania, w celu uzyskania potrzebnych do implementacji funkcjonalności. Może to wynikać z zadań już określonych (stan początkowy), lub dodanych w trakcie trwania sprintu.



### 2.8.2. Stany wejściowe dla stanu

- (opcjonalnie) brak - stan początkowy,
- TO\_DO,
- ON\_HOLD.

### 2.8.3. Stany wyjściowe dla stanu

- TO\_DO,
- ON\_HOLD,
- CANCELLED.

### 2.8.4. Kryteria wyjścia ze stanu

- gdy realizacja zadania jeszcze się nie rozpoczęła, a blokujące zadanie zostało zakończone (osiągnęło jeden ze stanów końcowych: TO\_DO lub CANCELLED), przechodzi ono do stanu TO\_DO,
- gdy realizacja zadania została przerwana, a blokujące zadanie zostało zakończone, przechodzi ono do stanu ON\_HOLD,
- gdy realizacja zadania w obecnym sprincie zostaje porzucona lub interesariusze rezygnują z implementacji w ostatecznym produkcie funkcjonalności realizowanej w ramach tego zadania, przechodzi ono do stanu CANCELLED.

## 2.9. CANCELLED

### 2.9.1. Definicja

Jeden z dwóch stanów końcowych (razem z TO\_DO). Realizacja zadania została porzucona - nie zostanie wykonane w obecnym sprincie lub w ogóle.

### 2.9.2. Stany wejściowe dla stanu

- TO\_DO,
- ON\_HOLD,
- BLOCKED.

### 2.9.3. Stany wyjściowe dla stanu

Brak - stan końcowy.

### 2.9.4. Kryteria wyjścia ze stanu

Brak - stan końcowy.

## 2.10. Diagram przejść pomiędzy stanami



### 2.10.1. Założenia

- elipsami oznaczane są stany początkowe,
- równoległobokami oznaczane są stany końcowe,
- stan BLOCKED nie jest oznaczony elipsą ze względu na jego opcjonalne bycie stanem początkowym.

### 2.10.2. Diagram przejść

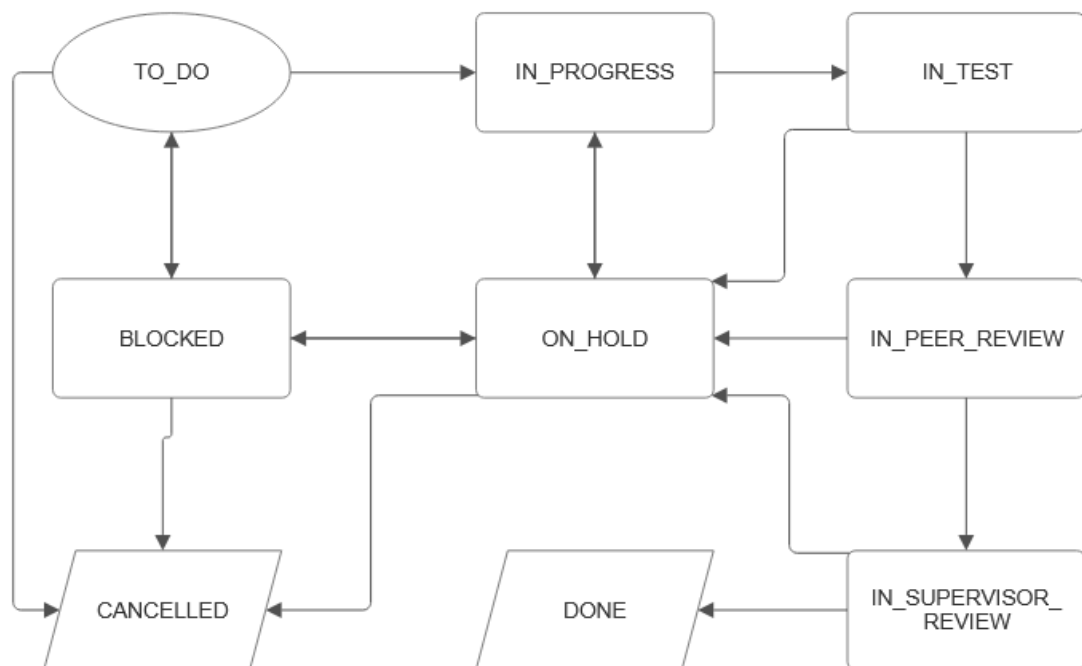


Fig 1. Diagram przejść pomiędzy stanami



## 3. Limity WIP

### 3.1. TO\_DO

Limit: brak limitu

Uzasadnienie: Kolumna TO\_DO służy jako backlog sprintu. Wszystkie zaplanowane zadania, na które zespół się umówił, mogą tu się znajdować. Nie ograniczamy liczby zgłoszeń, bo nie są one jeszcze realizowane.

### 3.2. IN\_PROGRESS

Limit: 3 zadania

Uzasadnienie: Każda osoba z zespołu może pracować nad jednym zadaniem w danym momencie. Limit 3 zapewnia, że każdy członek zespołu pracuje nad jednym zadaniem, nie zaczynając nowych, zanim nie zakończy obecnych. To minimalizuje przełączanie kontekstu i poprawia koncentrację.

### 3.3. ON\_HOLD

Limit: 3 zadania

Uzasadnienie: Jeśli zadania zostały wstrzymane, powinno ich być jak najmniej. Limit 3 pozwala na krótkoterminowe zatrzymanie pracy w wyjątkowych przypadkach (np. oczekiwanie na zależności), ale zapobiega zbieraniu się wielu nierozwiązanych spraw, które opóźniałyby projekt.

### 3.4. IN\_TEST

Limit: 1 zadanie

Uzasadnienie: Ponieważ testowaniem zajmuje się zwykle jedna osoba na raz (lub wymaga ono synchronizacji w zespole), ograniczenie do 1 zadanie zapewnia, że testowanie jest rozłożone w czasie i nie powoduje "wąskiego gardła".

### 3.5. IN\_PEER\_REVIEW

Limit: 1 zadanie

Uzasadnienie: Peer review wymaga dokładności i uwagi. Limit 1 pozwala skupić się na jednym przeglądzie na raz i nie powoduje zatorów, bo zespół jest mały (3 osoby).

### 3.6. IN\_SUPERVISOR\_REVIEW

Limit: 1 zadanie

Uzasadnienie: Ocena przez opiekuna projektu jest czynnością finalną przed uznaniem zadania za ukończone. Limit 1 zapewnia płynny przepływ i odzwierciedla fakt, że opiekun zwykle sprawdza zadania jedno po drugim.



### 3.7. DONE

Limit: brak limitu

Uzasadnienie: Zadania zakończone mogą być gromadzone bez ograniczeń.

### 3.8. BLOCKED

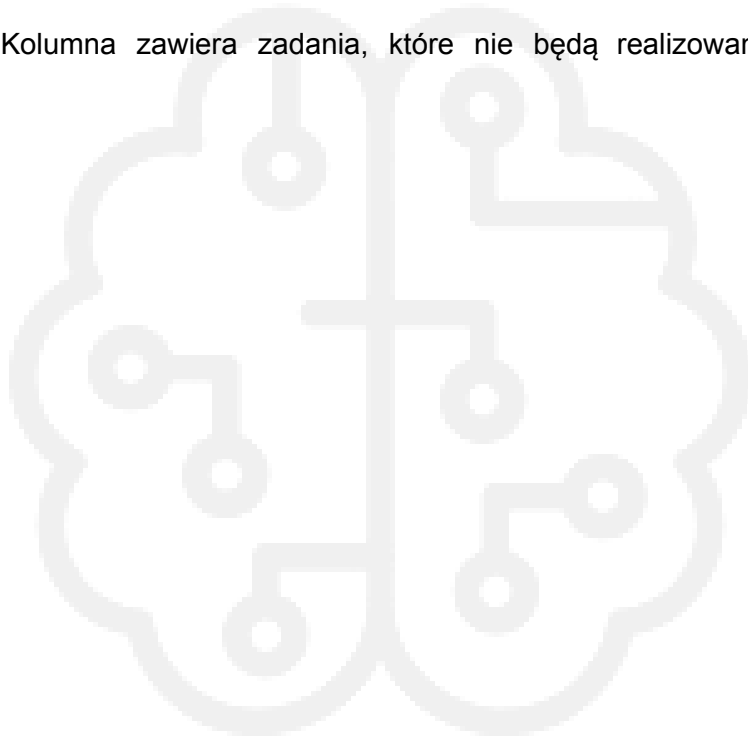
Limit: 2 zadania

Uzasadnienie: Zadania zablokowane powinny być priorytetowo rozwiązywane. Limit 2 wymusza szybkie reagowanie na blokery i minimalizuje ich liczbę w sprincie.

### 3.9. CANCELLED

Limit: brak limitu

Uzasadnienie: Kolumna zawiera zadania, które nie będą realizowane; nie wymaga limitowania.





## 4. Tablica Kanban

[Link to tablicy Kanban](#)

The screenshot shows a Kanban board with two columns: **TO\_DO** (3 items) and **BLOCKED** (2 / 2 items).

**TO\_DO Column:**

- Item 1:** Draft. -Zastąpienie aktualnego projektu pustym projektem. Priority: 3 (green), 1: MUST (red), 7h (grey), - (grey).
- Item 2:** Draft. -Możliwość w interfejsie graficznym istniejącego projektu na utworzenie nowego. Priority: 3 (green), 1: MUST (red), 2h (grey), - (grey).
- Item 3:** Draft. -Możliwość w interfejsie graficznym na wejściu programu opcji nowego projektu. Priority: 3 (green), 1: MUST (red), 2h (grey), - (grey).

**BLOCKED Column:**

- Item 1:** Draft. -Możliwość szybkiego zapisu poprzez ikonę interfejsu graficznego. Priority: 5 (yellow), 2: SHOULD (orange), 1.5h (grey), - (grey).
- Item 2:** Draft. -Możliwość zmiany wielkości okna przez złapanie za jego krawędź. Priority: 2 (green), 2: SHOULD (orange), 0.5h (grey), - (grey).

Each item has a green circle icon with a white 'D' and a pink circle icon with two vertical lines. At the bottom of each column is a '+ Add item' button.



IN\_PROGRESS

3 / 3

...

Zadanie aktywnie realizowane przez przypisanego członka zespołu.

RPI #21

Kreator nowych projektów

13

1:MUST

19.5h

19.5h

RPI #17

Szybki zapis do pliku

5

2:SHOULD

30h

30h

RPI #33

Rozszerzanie/zwężanie okienek

2

2:SHOULD

0.5h

0.5h

+ Add item

ON\_HOLD

2 / 3

...

Zadanie częściowo wykonane, aktualnie wstrzymane, bez przypisanego wykonawcy.

Draft

-Możliwość w interfejsie graficznym na wejściu programu wejścia w kreator projektu

13

1:MUST

2.5h

-

RPI #19

Wybór spośród 4 typów wykresów: dzwonowego, gaussowskiego, trójkątnego, trapezowego

2

1:MUST

9h

9h

+ Add item

Edytor graficzny systemów rozmytych dla języka Python

Dokument nr 4

Strona 15/20



○ IN\_PEER\_REVIEW 1/1 ...

Zadanie sprawdzane przez innego członka zespołu dla zapewnienia jakości i zgodności z wymaganiami.

Draft

-Możliwość szybkiego zapisu za pomocą skrótu klawiszowego

5 2:SHOULD 0.5h -

+ Add item

○ IN\_SUPERVISOR\_REVIEW 1/1 ...

Zadanie oceniane przez opiekuna projektu pod kątem zgodności ze wszystkimi wymaganiami.

Draft

-Wybór pomiędzy wnioskowaniem Mamdaniego oraz Sugeno

13 1:MUST 2h -

+ Add item





IN\_TEST 1 / 1

Zadanie w trakcie testowania pod kątem poprawności wykonania.

Draft

-Przeprowadzenie zapisu do pliku o domyślnym rozszerzeniu, w domyślnej lokalizacji

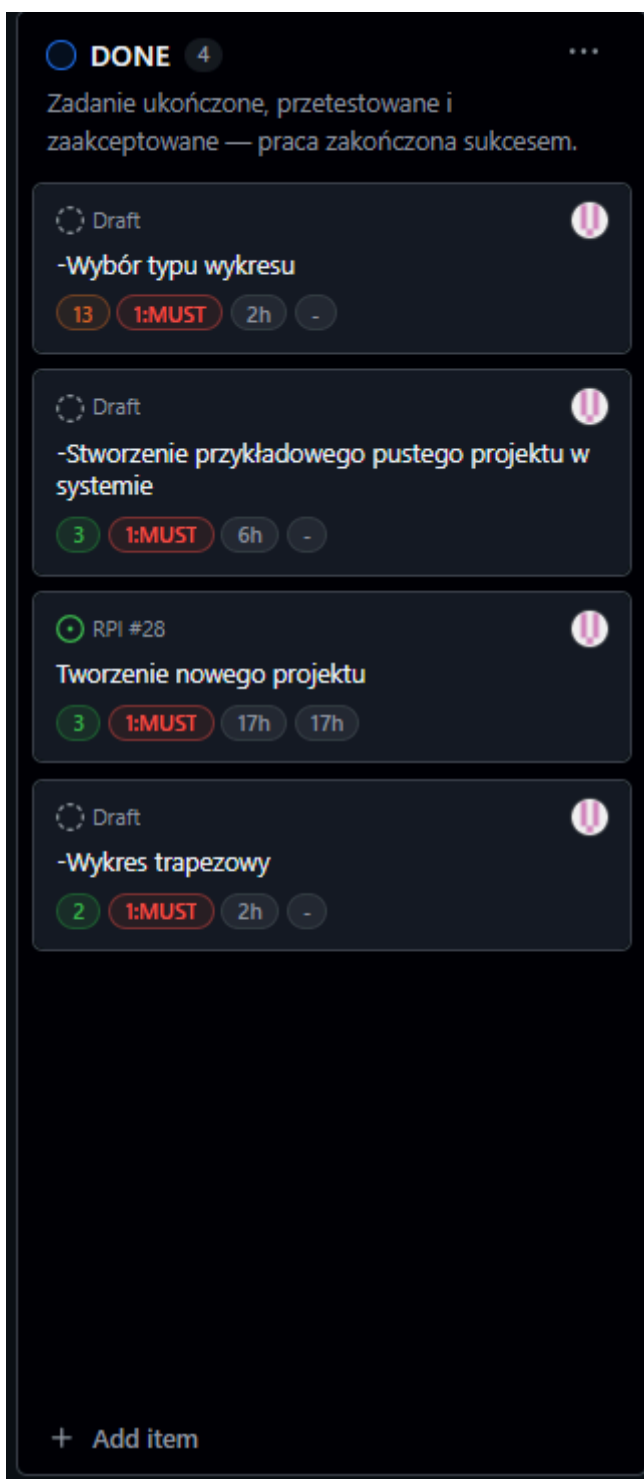
5 2:SHOULD 2h -

+ Add item

CANCELLED 0

Zadanie porzucone — nie będzie realizowane w tym sprincie ani później.

+ Add item





## 5. Metryki produktywności

### 5.1. Przepustowość

#### 5.1.1. Definicja

Przepustowość oznacza ilość pracy, która została zakończona w danym okresie czasu, na przykład jednego sprintu, tygodnia lub miesiąca. Za “pracę” zakończoną uznajemy zadanie, które przeszło od stanu TO\_DO do stanu DONE.

#### 5.1.2. Cel

- Miara produktywności i wydajności zespołu jako całości,
- Miara produktywności i wydajności indywidualnej członków zespołu,
- Podejmowanie decyzji odnośnie przyszłych harmonogramów i projektów.

### 5.2. Lead Time

#### 5.2.1. Definicja

Czas jaki upłynął od dodania zadania do tablicy Kanban do momentu oznaczenia go jako ukończone. Innymi słowy, oznacza ono łączny czas od zgłoszenia jaki musiał upłynąć, aby zadanie zostało faktycznie wykonane.

#### 5.2.2. Cel

- Precyzyjne planowanie,
- Efektywne przydzielanie zasobów do zadań,
- Optymalizacja cyklu pracy,
- Zapewnia kompleksowy obraz trwania zadań w systemie.

### 5.3. Długość cyklu

#### 5.3.1. Definicja

Czas cyklu jest częścią czasu realizacji, skupiającą się tylko i wyłącznie na czasie, w którym zespół aktywnie pracował nad zadaniem. Czas cyklu mierzy czas od oznaczenia zadania jako IN\_PROGRESS, aż do oznaczenia go jako DONE. Nie ma znaczenia ile czasu zajęło przejście ze stanu TO\_DO do stanu IN\_PROGRESS.

#### 5.3.2. Cel

- Miara wydajności zespołu,
- Optymalizacja cyklu aktywnej pracy zespołu.



## 5.4. Praca w toku (WIP)

### 5.4.1. Definicja

Praca w toku odnosi się do ilości zadań, nad którymi zespół “aktywnie” pracuje. Za zadania w toku uznajemy zadania w stanach IN\_PROGRESS, IN\_TEST, IN\_PEER\_REVIEW oraz IN\_SUPERVISOR\_REVIEW. Nie mierzy zadań, które zostały ukończone, ani tych które zostały wstrzymane lub jeszcze nie zaczęte.

### 5.4.2. Cel

- Identyfikacja wąskich gardeł w projekcie,
- Śledzenie i równoważenie obciążenia indywidualnych członków zespołu,
- Zarządzanie zaległościami i przeciążeniami.

## 5.5. Czas przebywania zadania w określonym stanie

### 5.5.1. Definicja

Przedstawia ile czasu każde zadanie spędziło w każdym z poszczególnych stanów. Pod uwagę bierzemy sumaryczną ilość czasu, nawet jeżeli wystąpił przypadek, w którym zadanie przechodziło kilkakrotnie w jeden stan. Nie bierzemy pod uwagę stanu DONE.

### 5.5.2. Cel

- Analiza stabilności i efektywności prac zespołu na każdym etapie,
- Dopasowanie zasobów do konkretnych etapów zadań,
- Podejmowanie decyzji odnośnie przepływu pracy.