



# PROJEKT INŻYNIERSKI

## Edytor graficzny systemów rozmytych dla języka Python

ID projektu - 46300

Opiekun projektu - dr inż. Jerzy Dembski

### Dokument nr 5: Scrum: Retrospektywa Sprintu

#### 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 przeprowadzenie retrospektywy sprintu w wybranym projekcie z użyciem konkretnej wybranej techniki lub gry zespołowej, a także spisanie wyników i wniosków z tej retrospektywy.

Wersja:	1.2
Data wydania:	19.05.2025
Redaktor:	Adam Zarzycki
Współautorzy:	Filip Wesołowski, Julian Kulikowski
Etap/zadanie:	5
Nazwa pliku:	Retrospektywa_Sprintu_v1.2.docx
Liczba stron:	10



## Historia zmian

Wersja	Data	Opis zmiany
1.0	19.05.2025	Dodanie punktów 1. i 2.
1.1	20.5.2025	Dodanie punktów 3. i 4.
1.2	22.5.2025	Dodanie punktu 5.





## SPIS TREŚCI

<b>1. O projekcie i produkcji</b>	<b>4</b>
<b>2. Wybrana technika retrospektywy</b>	<b>4</b>
2.1. Wybór	4
2.2. Opis techniki	4
2.3. Ilustracja działania techniki	5
2.4. Uzasadnienie wyboru techniki	6
2.5. Źródła wiedzy o technice	6
<b>3. Przebieg i wyniki retrospektywy</b>	<b>6</b>
3.1. Przebieg retrospektywy	6
3.1.1. Czas i miejsce	6
3.1.2. Uczestnicy i ich role	6
3.1.3. Wykorzystane narzędzia	7
3.2. Wyniki retrospektywy	7
3.2.1. Wydobyte informacje	7
3.2.2. Możliwe rozwiązania	8
<b>4. Zadania do wykonania - action items</b>	<b>8</b>
4.1. Późne spotkania organizacyjne	9
4.2. Mapowanie zależności	9
4.3. Słabe szacowanie złożoności zadań	9
<b>5. Wnioski</b>	<b>9</b>



# 1.0 projekcie i produkcji

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. W 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. Wybrana technika retrospektywy

### 2.1. Wybór

W ramach stosowanej techniki retrospektywy, nasz zespół projektowy wybrał technikę KALM.

### 2.2. Opis techniki

KALM (Keep, Add, Less, More) to technika skupiająca się na ocenie wartości poszczególnych elementów poprzedniego sprintu. Pewną tablicę (fizyczną lub cyfrową) dzieli się na 4 obszary, zgodnie z hasłami zawartymi w samej nazwie techniki:

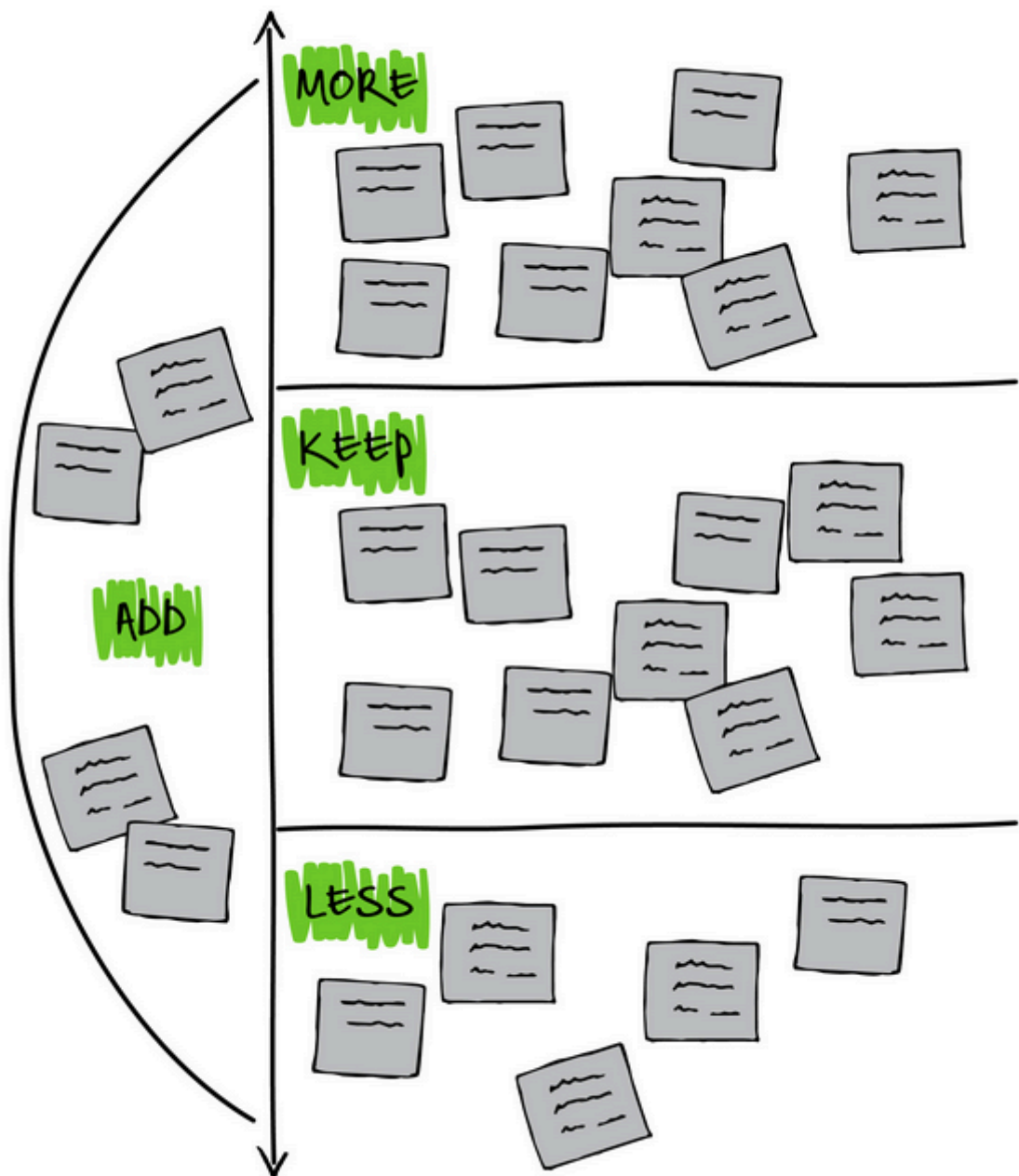
- **Keep** - elementy, które chciałbyś zachować na następny sprint,
- **Add** - nowe pomysły, które mielibyśmy wykorzystać w następnym sprincie,
- **Less** - jakie elementy sprintu uważasz za niepotrzebne / nieproduktywne / niepotrzebne, i chciałbyś usunąć je z następnego sprintu lub ograniczyć,
- **More** - jakie elementy uważasz za korzystne i chciałbyś je rozwinąć w następnym sprincie.

Następnie zespół projektowy jest proszony o przyporządkowanie elementów poprzedniego sprintu do ww. kategorii (Keep, Less, More), lub dodać swoje własne pomysły na następny sprint (Add). Na koniec przeprowadzana jest dyskusja na podstawie uzyskanych wyników, jak również wybierane Action Points

(AP) na następny sprint.

Opcjonalnie, w zależności od charakteru, doświadczenia i nastrojów panujących w zespole, można wprowadzić anonimowość podawania opinii lub elementy brainwritingu, by zwiększyć zaangażowanie, poczucie bezpieczeństwa, i bezstronność uczestników spotkania.

## 2.3. Ilustracja działania techniki



Rys 1. Ilustracja podziału tablicy KALM, zaczerpnięta ze źródeł



## 2.4. Uzasadnienie wyboru techniki

Czynniki, które wpłynęły na nasz wybór:

- wizualizacja retrospektywy w formie graficznej na tablicy (nie tylko o tym mówimy, można to również zobaczyć),
- wizualizacja ilościowa przydziału poszczególnych elementów na tablicy (widzimy, gdy jedna z kategorii jest zdecydowanie bardziej zapełniona niż inne, np. przodowanie kategorii Less może nam zasugerować zmiany w samym planie sprintu),
- pomysły na nowe elementy mogą być anonimowe (nie ma presji na konkretną osobę w przypadku niepowodzenia danego pomysłu),
- wyróżnienie kategorii Add dla nowych pomysłów, promujące wychodzenie z nowymi inicjatywami przez członków zespołu projektowego, w ramach poszerzania wiedzy o zadaniu (projekcie) i zdobywania nowych doświadczeń (np. podczas seminariów lub spotkań z opiekunem projektu),
- wg wykładu dra inż. Jakuba Milera, technika KALM jest bardzo wysoko oceniana (4.05) przez osoby bez doświadczenia w przeprowadzaniu retrospektywy.

## 2.5. Źródła wiedzy o technice

- <https://www.funretrospectives.com/kalm-keep-add-more-less/>,
- <https://www.teamretro.com/retrospectives/kalm-retrospective>,
- wykład nr 4 dra inż. Jakuba Milera.

# 3. Przebieg i wyniki retrospektywy

## 3.1. Przebieg retrospektywy

### 3.1.1. Czas i miejsce

Retrospektywa miała miejsce po zakończeniu pierwszego dwutygodniowego sprintu zespołu dnia 14.05.2025. Przyjęła ona formę spotkania zdalnego na platformie Discord. Całość retrospektywy sprintu trwała półtorej godziny.

### 3.1.2. Uczestnicy i ich role

W spotkaniu brał udział cały zespół Scrum. Każdy z uczestników przyjął kluczową dla retrospektywy rolę:

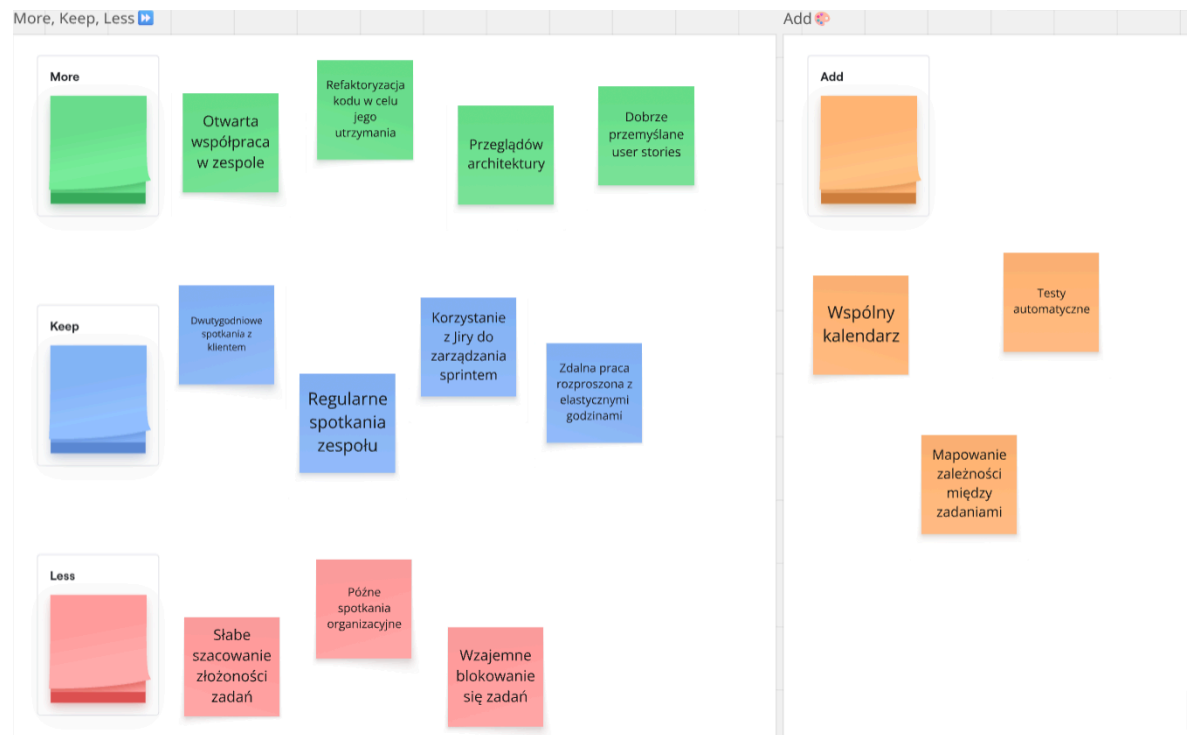
- **Adam Zarzycki** - odpowiedzialny za notowanie informacji wydobytych w trakcie retrospektywy, a także jej przebiegu,
- **Filip Wesołowski** - moderator odpowiedzialny za przebieg retrospektywy oraz pilnujący, by odbyła się zgodnie z techniką KALM,
- **Julian Kulikowski** - osoba odpowiedzialna za techniczne aspekty retrospektywy, przygotowująca tablicę KALM.



### 3.1.3. Wykorzystane narzędzia

- **OneNote** - narzędzie firmy Microsoft pozwalające na prowadzenie zdalnego notesu oraz udostępnianie notatek z retrospektywy członkom zespołu,
- **Discord** - zdalna platforma komunikacyjna z pomocą, której zespół przeprowadził retrospektywę,
- **Jira** - platforma umożliwiająca prowadzenie zdalnego backlogu produktu oraz backlogu sprintu, wykorzystywana do dokumentacji przebiegu sprintu,
- **Miro** - platforma służąca do rozproszonego zarządzania projektami, wykorzystana jako tablica KALM.

## 3.2. Wyniki retrospektywy



Rys 2. Powstała w wyniku retrospektywy tablica KALM

### 3.2.1. Wydobyte informacje

Najważniejszą informacją wydobytą w wyniku retrospektywy było ogólne zadowolenie zespołu z przyjęcia metodyk zwinnych do projektu. Rozproszony tryb pracy również został uznany za pozytywny aspekt. Zespół wyraził szczególne zadowolenie z zewnętrznego zgrania, dzielenia się wiedzą oraz współpracą. Jako kolejny pozytywny aspekt zostały wspomniane częste spotkania z klientem w celu kontroli wymagań, zakresu projektu oraz kontroli jakości, zespół stwierdził jednak, iż aktualny tryb współpracy z klientem jest odpowiadający i nie ma potrzeby na częstsze spotkania. Finalnym wartym uwagi pozytywnym aspektem wspomnianym na spotkaniu były przeglądy



wykonywane architektury oraz refaktoryzacja wykorzystywane w celu ujednolicenia projektu oraz zapewnienia jakości.

Największym problemem zespołu okazały się do tej pory późne godziny przeprowadzanych spotkań, które ze względu na dostępność członków odbywają się dopiero pod wieczór. Sprawia to, iż członkowie zespołu często przychodzą na nie zmęczeni i bardziej są zainteresowani ich końcem, aniżeli przebiegiem. Innym aspektem, który wymaga poprawy, jest słabe oszacowanie zadań w backlogu pod względem czasochłonności godzinowej oraz story points. Wiele zadań okazało się przeszacowanych lub niedocenionych. Finalną poruszoną kwestią było blokowanie się wzajemnie członków zespołu ze względu na zależność zadań od siebie.

Zespół wyraził chęć dodania dodatkowych narzędzi do projektu typu kalendarz w celu lepszej kontroli dostępności członków. Zaproponowano również, aby dodać mapę zależności komponentów oraz testy automatyczne.

### 3.2.2. Możliwe rozwiązania

Problem: Słabe szacowanie złożoności zadań

- Ponowny przydział Story Points do zadań,
- Zmiana sposobu przydzielania złożoności zadań.

Problem: Późne spotkania organizacyjne

- Znaleźnienie alternatywnego terminu spotkań,
- Kontrolowanie długości spotkań,
- Podział dłuższego spotkania na kilka krótszych,
- Nagrywanie spotkania dla nieobecnych,
- Asynchroniczne spotkania zespołu.

Problem: Wzajemne blokowanie się zadań

- Wprowadzenie przeglądów zależności,
- Lepsze ustalenie priorytetów zadań,
- Przegląd kalendarza dostępności członków zespołu podczas podziału zadań,
- Wyznaczanie terminów dla konkretnych zadań.

## 4. Zadania do wykonania - action items

W wyniku retrospektywy postanowiono zebrać zadania z kategorii "Less" oraz "Add" i trzy z nich zaimplementować w ramach następnego sprintu. Wybór zadań zespół zdecydował się przeprowadzić za pomocą przydzielenia każdemu z nich punkty priorytetu od 1 do 5 według własnej oceny, gdzie 1 to najniższy priorytet dla zadań mniej ważnych, a 5 najwyższy. Priorytety te następnie zsumowano i wybrano trzy zadania z najwyższą sumą.





## 4.1. Późne spotkania organizacyjne

Punkty priorytetu:

- Filip: 5
- Julian: 4
- Adam: 4

Suma punktów priorytetu: 13

Zespół uznał, iż bardzo ważnym aspektem mającym wpływ na produktywność jest godzina spotkań. Zdecydowano się na ponowny przegląd grafików członków zespołu w celu sprawdzenia czy od poprzedniego sprintu nie nastąpiły zmiany. Po wspólnym przeglądzie terminów i dat wyznaczono nową godzinę spotkań, która będzie obowiązywać od następnego sprintu.

## 4.2. Mapowanie zależności

- Filip: 3
- Julian: 5
- Adam: 4

Suma punktów priorytetu: 12

Następnym zadaniem, które zespół wyznaczył w celu poprawy pracy jest utworzenie mapy zależności konkretnych komponentów systemu. Ma ono pozwolić na lepszą kontrolę jakości, łatwiejszą implementację kolejnych elementów backlogu oraz łatwiejsze wyznaczanie priorytetów. Zespół w celu realizacji tego zadania umówił się na spotkanie, podczas którego wykona diagram UML dla programu.

## 4.3. Słabe szacowanie złożoności zadań

- Filip: 4
- Julian: 2
- Adam: 5

Suma punktów priorytetu: 11

Ostatnim zadaniem realizowanym retrospektywy sprintu jest poprawa szacowania złożoności zadań. Zespół bogatszy o doświadczenie po pierwszym sprincie uznał, iż trafniejsze oddanie złożoności zadań będzie pomocne przy ich podziale i ustalaniu priorytetów. W tym celu zdecydowano się na ponowne rozdysponowanie Story Points dla elementów backlogu produktu za pomocą metody playing poker na początku następnego spotkania.

# 5. Wnioski

Zastosowanie techniki KALM w przeprowadzeniu retrospektywy zostało przez zespół ocenione jako skuteczne i satysfakcjonujące. Sama technika okazała się łatwa do przyswojenia nawet dla osób bez wcześniejszego doświadczenia w retrospektywach. Jej graficzny charakter, możliwość anonimizacji wypowiedzi oraz jasny podział na cztery proste kategorie sprzyjały aktywnemu udziałowi wszystkich członków zespołu.



KALM umożliwiło nie tylko efektywne zidentyfikowanie problemów i pozytywnych aspektów sprintu, ale również zachęciło zespół do zaproponowania nowych inicjatyw, co przełożyło się na konkretne action items do wdrożenia w kolejnym sprincie. Wizualizacja ilościowa pozwoliła na szybkie wychwycenie dominujących tematów, co ułatwiło późniejszą dyskusję.

Z perspektywy celu retrospektywy – czyli refleksji nad przebiegiem sprintu, wyciągnięcia wniosków oraz zaplanowania usprawnień – technika okazała się bardzo skuteczna. Uczestnicy wyrazili zadowolenie z formy i przebiegu spotkania, a sama retrospektywa przebiegła w przyjaznej i konstruktywnej atmosferze. W związku z tym zespół rozważa kontynuację stosowania tej techniki również w kolejnych sprintach.

