



PROJEKT INŻYNIERSKI

Edytor graficzny systemów rozmytych dla języka Python

ID projektu - 46300

Opiekun projektu - dr inż. Jerzy Dembski

Dokument nr 3: Scrum: Backlog 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 zadania jest opracowanie backlogu wybranego sprintu tak, jakby się to robiło podczas scrumowego spotkania planowania sprintu.

Wersja:	1.2
Data wydania:	12.04.2025
Redaktor:	Adam Zarzycki
Współautorzy:	Filip Wesołowski, Julian Kulikowski
Etap/zadanie:	3
Nazwa pliku:	ScrumBS_v1.1.docx
Liczba stron:	23



Historia zmian

Wersja	Data	Opis zmiany
1.0	12.04.2025	Dodanie punktów 1., 2. i 3.
1.1	13.04.2025	Dodanie punktów 4. i 5.
1.2	15.04.2025	Dodanie punktów 6., 7. i 8.





SPIS TREŚCI

1. O projekcie i produkcji	4
2. Oszacowanie rozmiaru Backlogu Produktu	5
2.1. Zastosowana metoda	5
2.2. Sposób przydziału punktów	5
2.3. Przebieg spotkania	5
2.3.1. Przygotowanie	5
2.3.2. Założenia	5
2.3.3. Głosowania	5
2.4. Udokumentowanie przebiegu spotkania	6
2.5. Stan Backlogu Projektu po spotkaniu	11
3. Przygotowanie Backlogu Sprintu	12
3.1. Założenia	12
3.1.1. Sprint	12
3.1.2. Pojemność zespołu	12
3.2. Dobór zakresu sprintu	13
3.2.1. Sposób wyboru elementów do sprintu	13
3.2.2. Wybrane elementy Backlogu Produktu	13
3.2.3. Uzasadnienie wyboru elementów	13
3.2.4. Oszacowanie godzinowe wybranych elementów	13
3.2.5. Uzasadnienie oszacowania godzinowego wybranych elementów	14
3.3. Podsumowanie	14
4. Cel sprintu	15
5. Backlog sprintu	15
6. Kryteria akceptacji	16
6.1. Tworzenie nowego projektu	16
6.2. Wnioskowanie Mamdaniego i Sugeno po kroku	17
6.3. Definiowanie zmiennych i reguł rozmytych	18
6.4. Wybór spośród 4 typów wykresów: dzwonowego, gaussowskiego, trójkątnego, trapezowego	19
6.5. Rozszerzanie/zwężanie okienek	20
6.6. Kreator nowych projektów	21
6.7. Szybki zapis do pliku	22
7. Definicja ukończenia (DoD)	23
8. Bibliografia	23

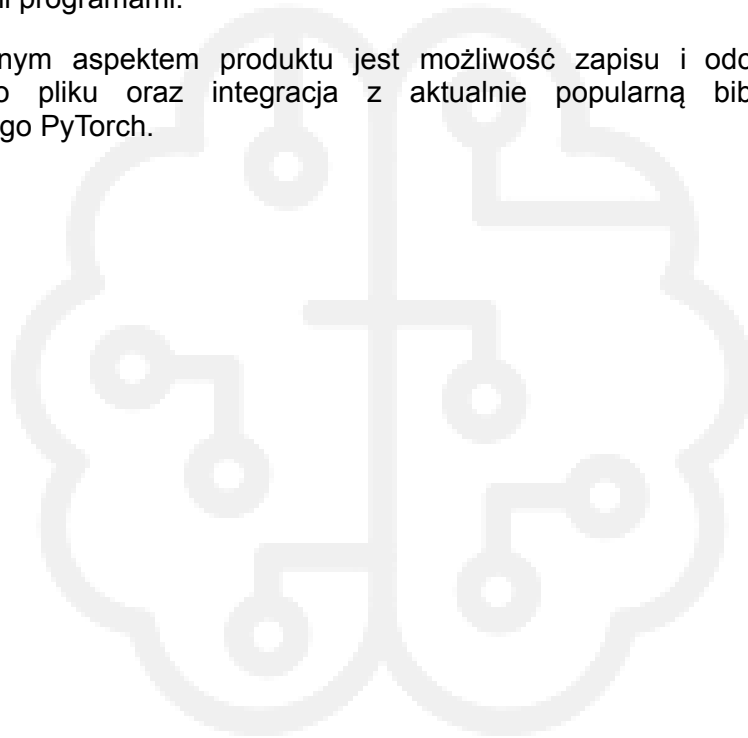


1. O projekcie i produkcji

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. Oszacowanie rozmiaru Backlogu Produktu

2.1. Zastosowana metoda

Do oszacowania rozmiaru elementów Backlogu Produktu (dalej: elementów BP) zastosowano metodę Story Points (dalej: SP). Jako wartości przydzielanych SP wybrano te wg skali M. Cohna:

{1, 2, 3, 5, 8, 13, 20, 40, 100}

2.2. Sposób przydziału punktów

Punkty SP przydzielane były zgodnie techniką Planning Poker. W tym celu grupa projektowa spotkała się w pełnym składzie na spotkaniu, podczas którego przeprowadzone zostało głosowanie nad złożonością każdego z elementów Backlogu Produktu.

2.3. Przebieg spotkania

2.3.1. Przygotowanie

Przed rozpoczęciem spotkania, do wykorzystywanej aplikacji (link w Bibliografii) dodany został zbiór kart zgodny ze skalą M. Cohna. Następnie utworzona została nowa gra, do której przepisane zostały elementy Backlogu Produktu (zgodnie ze stanem na dzień 09.04.2025), z dokładnością do nazwy elementu BP.

2.3.2. Założenia

Julian Kulikowski został wybrany na prowadzącego spotkanie; jego zadania sprowadzały się do wyboru kolejnego elementu głosowania i oznajmiania tego reszcie obecnych.

Elementy BP poddawane były głosowaniu w kolejności losowej, by zapobiec stronniczości wywołanej posiadaniem przez elementy BP wstępnego przypisania złożoności w SP (wykonanego w ramach etapu 2. projektu).

2.3.3. Głosowania

Głosowania przeprowadzane były według następującego schematu:

- prowadzący spotkanie wybierał kolejny element BP pod głosowanie i oznajmiał to wszystkim obecnym,
- głosujący wybierali kartę z preferowanym oszacowaniem,
- po oddaniu głosów przez wszystkich obecnych, system pokazywał wartości wybranych kart reszcie głosujących:
 - w przypadku jednomyślności głosujących, elementowi BP nadawana była wybrana wartość SP,



- w przeciwnym przypadku, głosujący mogli wyrazić swoją opinię na temat złożoności kontrowersyjnego elementu BP innym głosującym,
- następnie przeprowadzane było ponowne głosowanie:
 - jeżeli tym razem decyzja była jednomyślna, element BP otrzymywał takie dopasowanie,
 - w przeciwnym wypadku, element BP otrzymywał dopasowanie zgodne z opinią większości głosujących,
 - w sytuacji wystąpienia trzech różnych głosów, decyzję podejmował prowadzący spotkanie,
- prowadzący spotkanie oznajmiał wszystkim obecnym ostateczny przydział SP do elementu BP,
- nowa wartość SP wprowadzana była do Backlogu Produktu.

Proces ten powtórzony został dla wszystkich elementów BP.

2.4. Udokumentowanie przebiegu spotkania

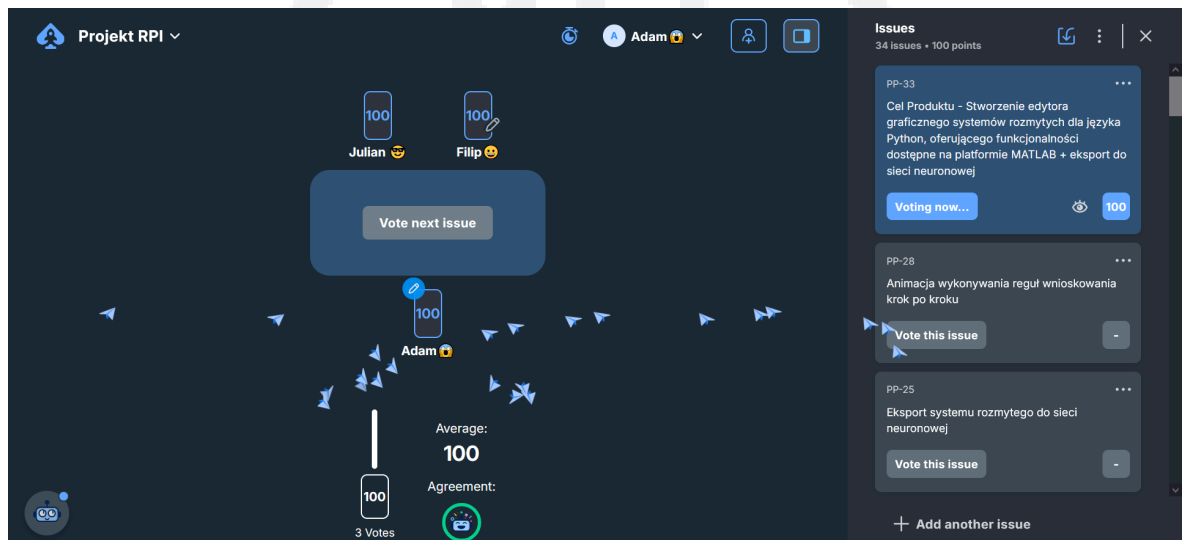


Fig 1. Głosowanie nad złożonością Celu Produktu - pełna zgoda

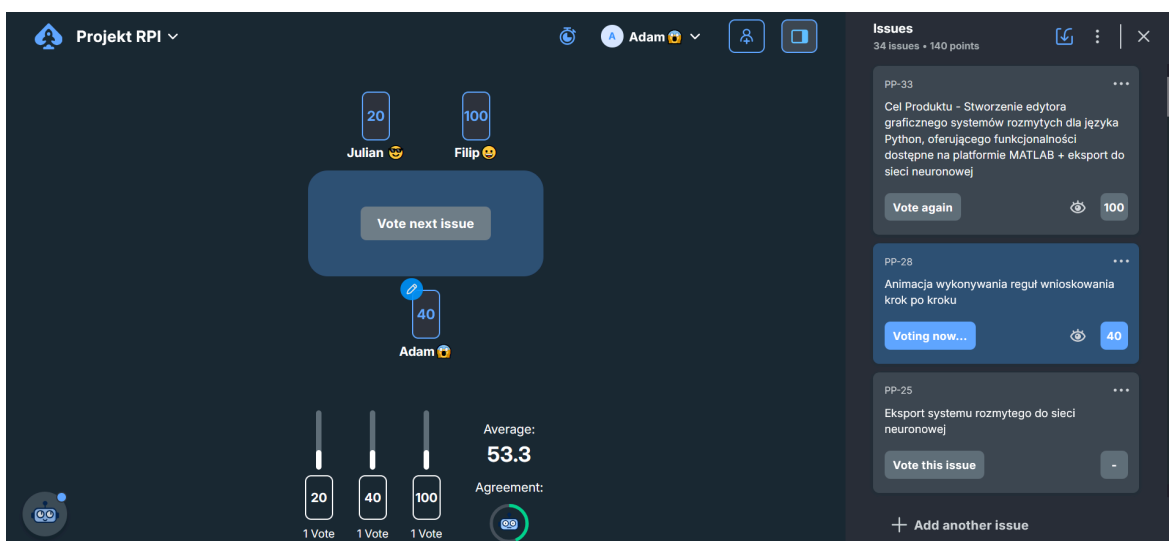


Fig 2. Głosowanie nad elementem BP - 3 różne opinie

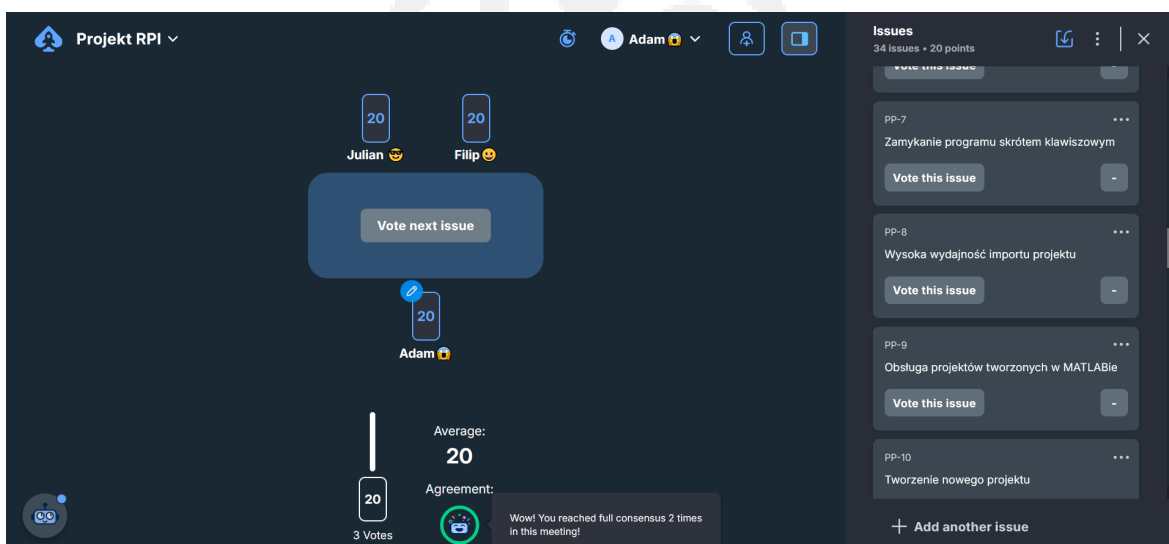


Fig 3. Ponowne głosowanie nad elementem BP z Fig 2. - pełna zgoda

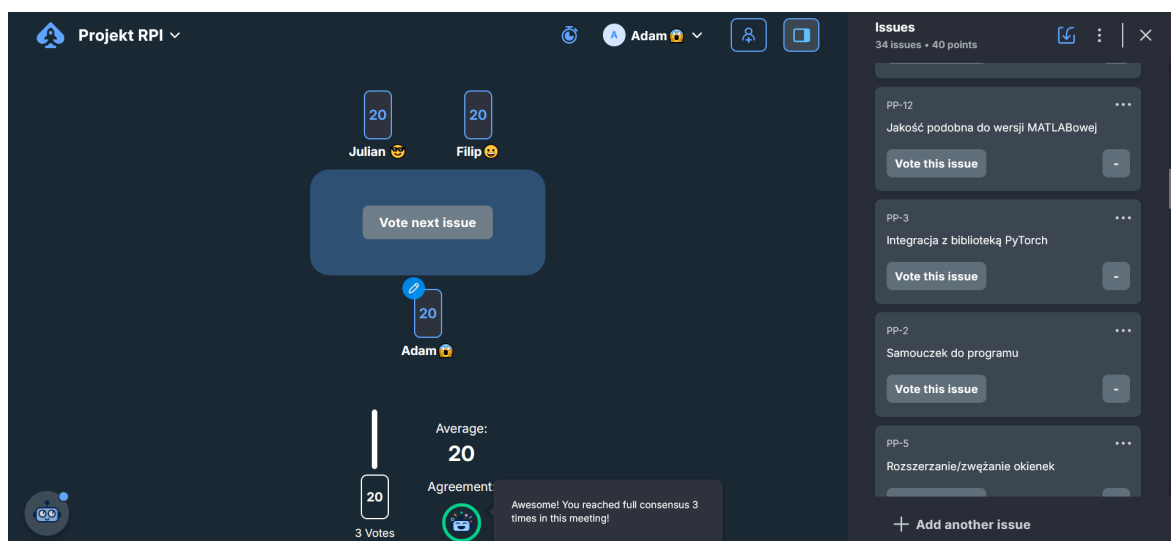


Fig 4. Głosowanie nad elementem Bp - pełna zgoda

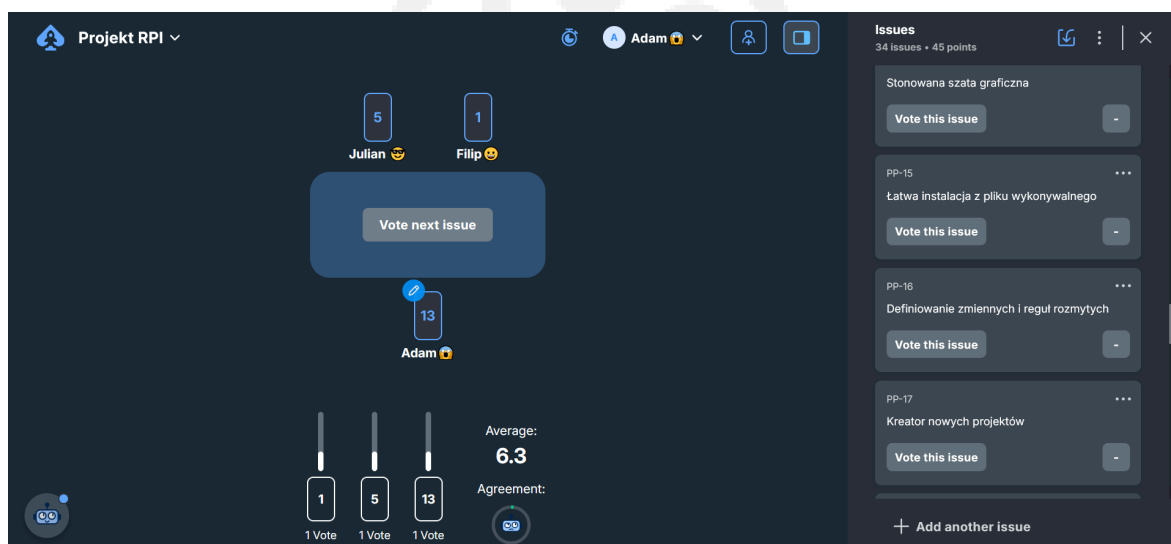


Fig 5. Głosowanie nad elementem BP - 3 różne opinie

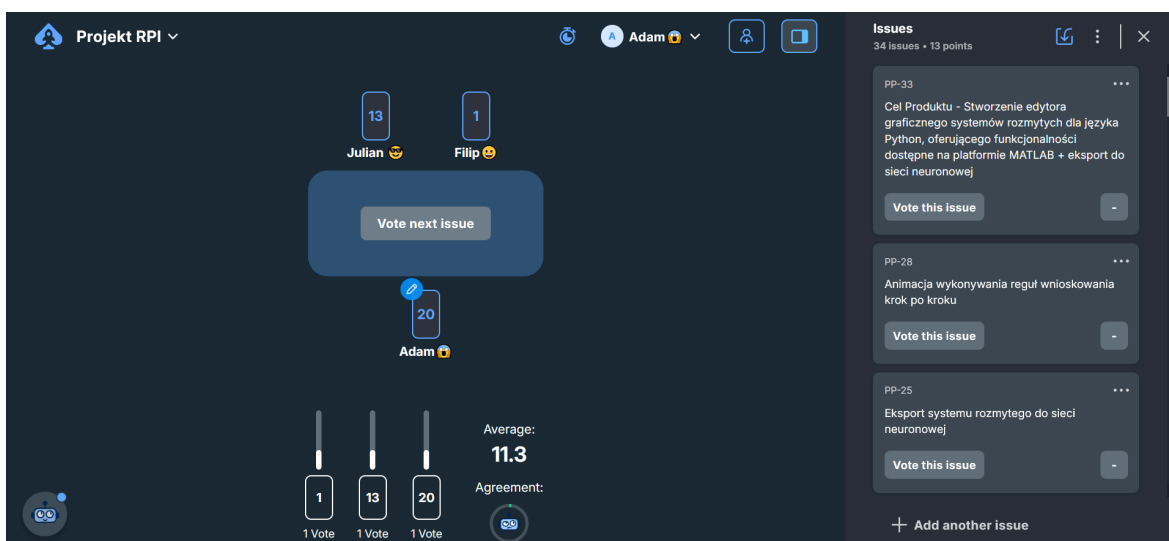


Fig 6. Ponowne głosowanie nad elementem BP z Fig. 5 - znowu 3 różne opinie, wybrana została opinia prowadzącego spotkanie (Juliana Kulikowskiego)

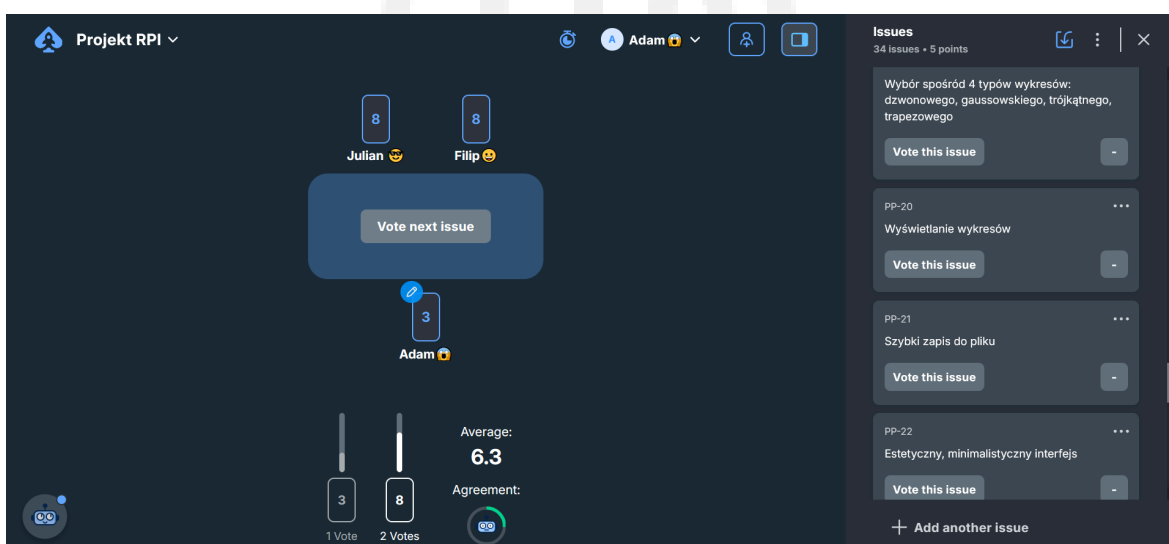


Fig 7. Ponowne Głosowanie nad pewnym elementem Bp - sytuacja 2:1, zwyciężyła opinia większości

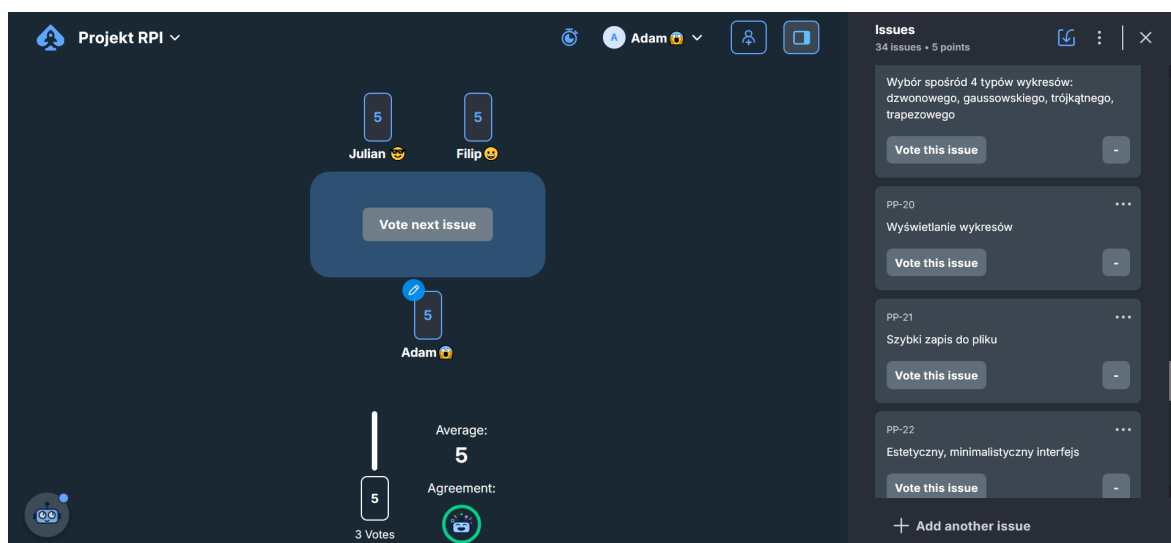


Fig 8. Głosowanie nad elementem BP - pełna zgoda

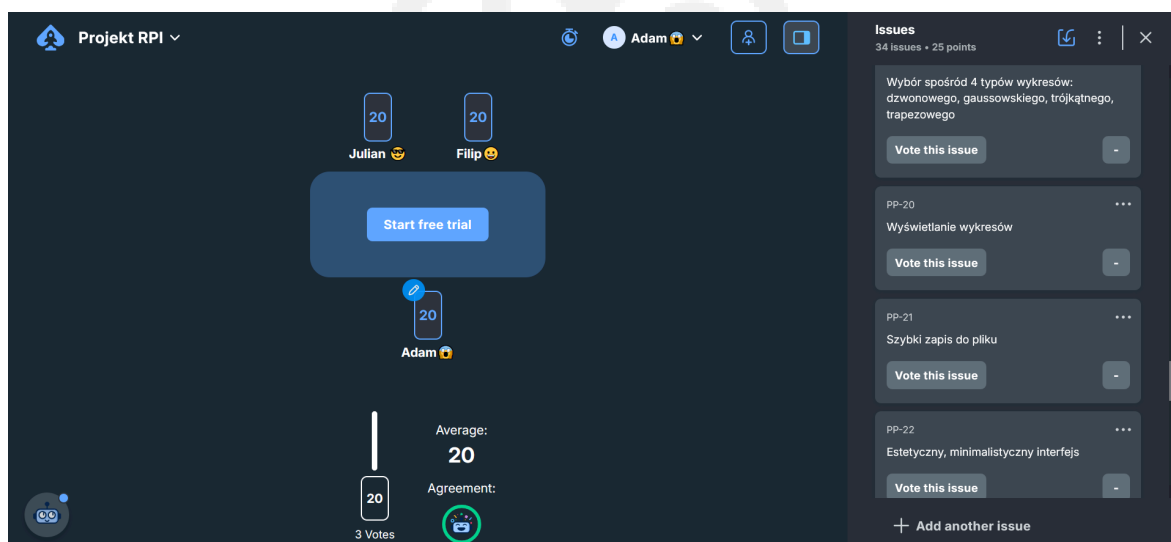


Fig 9. Głosowanie nad elementem BP - pełna zgoda



2.5. Stan Backlogu Projektu po spotkaniu

1	🕒 Cel Produktu - Stworzenie edytora graficznego systemów rozmytych dla języka Python, ofer... #5	100	▼	1:MUST	▼
2	🕒 Eksport systemu rozmytego do pliku #4	20	▼	1:MUST	▼
3	🕒 Import systemu rozmytego z pliku, w tym z pliku .zip #6	20	▼	1:MUST	▼
4	🕒 Animacja wykonywania reguł wnioskowania krok po kroku #10	20	▼	1:MUST	▼
5	🕒 Edycja wykresów poprzez przeciąganie na ekranie #9	13	▼	1:MUST	▼
6	🕒 Modyfikacja wartości zmiennych w trakcie wykonywania wnioskowania #11	13	▼	1:MUST	▼
7	🕒 Jakość podobna do wersji MATLABowej #26	13	▼	1:MUST	▼
8	🕒 Wnioskowanie Mamdaniego i Sugeno #20	8	▼	1:MUST	▼
9	🕒 Definiowanie zmiennych i reguł rozmytych #22	8	▼	1:MUST	▼
10	🕒 Obsługa projektów tworzonych w MATLABie #29	8	▼	1:MUST	▼
11	🕒 Zapisywanie zmian #15	5	▼	1:MUST	▼
12	🕒 Tworzenie nowego projektu #28	3	▼	1:MUST	▼
13	🕒 Wyświetlanie wykresów #18	2	▼	1:MUST	▼
14	🕒 Wybór spośród 4 typów wykresów: dzwonowego, gaussowskiego, trójkątnego, trapezowego #19	2	▼	1:MUST	▼
15	🕒 Kreator nowych projektów #21	13	▼	2:SHOULD	▼
16	🕒 Wysoka wydajność importu projektu #30	13	▼	2:SHOULD	▼
17	🕒 Szybki zapis do pliku #17	5	▼	2:SHOULD	▼
18	🕒 Minimalizacja zmian interfejsu względem interfejsu MATLABowego #27	3	▼	2:SHOULD	▼
19	🕒 Intuicyjne menu #37	3	▼	2:SHOULD	▼
20	🕒 Możliwość ustawiania/przesuwania okien na ekranie #8	2	▼	2:SHOULD	▼
21	🕒 Pomoc dla użytkownika #25	2	▼	2:SHOULD	▼
22	🕒 Rozszerzanie/zwązanie okienek #33	2	▼	2:SHOULD	▼
23	🕒 Tryb jasny/ciemny #7	1	▼	2:SHOULD	▼
24	🕒 Dodanie skrótów klawiszowych #12	1	▼	2:SHOULD	▼
25	🕒 Estetyczny, minimalistyczny interfejs #16	1	▼	2:SHOULD	▼
26	🕒 Łatwa instalacja z pliku wykonywalnego #23	40	▼	3:COULD	▼
27	🕒 Eksport systemu rozmytego do sieci neuronowej #13	20	▼	3:COULD	▼
28	🕒 Import systemu rozmytego z sieci neuronowej #14	20	▼	3:COULD	▼
29	🕒 Integracja z biblioteką PyTorch #35	20	▼	3:COULD	▼
30	🕒 Zamykanie programu skrótem klawiszowym #31	2	▼	3:COULD	▼
31	🕒 Samouczek do programu #36	2	▼	3:COULD	▼
32	🕒 Stonowana szata graficzna #24	1	▼	3:COULD	▼
33	🕒 Dostosowywanie kolorów interfejsu #32	1	▼	3:COULD	▼
34	🕒 Automatyczna generacja reguł rozmytych #34	20	▼	4:WONT	▼



3. Przygotowanie Backlogu Sprintu

3.1. Założenia

3.1.1. Sprint

- *start sprintu* - 16.04.2025,
- *koniec sprintu* - 30.04.2025,
- *czas na projekt* = 14 tygodni (6 tygodni 6. semestru + 8 tygodni 7. semestru (2 ostatnie tygodnie przeznaczone są na zatwierdzenie pracy przez recenzenta),
- *długość sprintu* = 2 tygodnie,
- *liczba sprintów* = $\text{czas na projekt} / \text{długość sprintu} = 7$,
- *suma SP* = 287 (bez Celu Produktu),
- *średnia liczba SP na sprint* = $\text{suma SP} / \text{liczba sprintów} = 41$ [SP/sprint],
- *zakładana średnia prędkość sprintu* = $\text{zaokr_w_górę}(\text{średnia liczba SP na sprint}) = 41$,
- *zakres na pierwszy sprint* = zakładana średnia prędkość = 41.

3.1.2. Pojemność zespołu

	Czas całkowity	Czas wyłączony ze sprintu			
	Typowy czas pracy [h]	Dzień ustawowo wolny [h]	Wyjazdy [h]	Spotkania [h]	"Szum" [%]
Adam Zarzycki	80	24	8	2	20
Julian Kulikowski	80	24	0	10	20
POZOSTAŁO					



Filip Wesołowski	80	24	3	5	20	
RAZEM	240	72	11	17	20	<u>112</u>

3.2. Dobór zakresu sprintu

3.2.1. Sposób wyboru elementów do sprintu

Planowany w tym dokumencie sprint jest pierwszym w tym projekcie, w związku z czym elementy wybierano głównie z myślą o ich podstawowym znaczeniu dla całego projektu.

Zespół projektowy uzgodnił zakres sprintu po konsultacji z Właścicielem Produktu (w ramach tego projektu funkcję tę pełni Adam Zarzycki).

3.2.2. Wybrane elementy Backlogu Produktu

- #12 - Tworzenie nowego projektu, 3 SP,
- #20 - Wnioskowanie Mamdaniego i Sugeno, 8 SP,
- #22 - Definiowanie zmiennych i reguł rozmytych, 8 SP,
- #19 - Wybór spośród 4 typów wykresów: dzwonowego, gaussowskiego, trójkątnego, trapezowego, 2 SP,
- #33 - Rozszerzanie/zwężanie okienek, 2 SP,
- #21 - Kreator nowych projektów, 13 SP,
- #17 - Szybki zapis do pliku, 5 SP.

RAZEM - 41 SP.

3.2.3. Uzasadnienie wyboru elementów

Podane wyżej elementy wybrane zostały ze względu na ich podstawowe znaczenie dla dalszego rozwoju programu. Większość reszty wymaganych funkcjonalności albo wymaga tychże elementów do poprawnego funkcjonowania, albo we fragmencie lub w całości od nich zależy. Ponadto, elementy te pozwalają wytworzyć szkielet programu do pokazania interesariuszom projektu.

3.2.4. Oszacowanie godzinowe wybranych elementów

- #12 - Tworzenie nowego projektu, 17 h,
- #20 - Wnioskowanie Mamdaniego i Sugeno, 27 h,
- #22 - Definiowanie zmiennych i reguł rozmytych, 6 h,



- #19 - Wybór spośród 4 typów wykresów: dzwonowego, gaussowskiego, trójkątnego, trapezowego, 9 h,
- #33 - Rozszerzanie/zwężanie okienek, 0,5 h,
- #21 - Kreator nowych projektów, 19,5 h,
- #17 - Szybki zapis do pliku, 30 h.

RAZEM - 109h.

3.2.5. Uzasadnienie oszacowania godzinowego wybranych elementów

Najbardziej czasochłonnymi elementami tego sprintu będą “Wnioskowanie Mamdaniego i Sugeno” (ze względu na potrzebę dogłębnego zapoznania się z dostępną literaturą naukową i szczegółowego zrozumienia tematu) i “Szybki zapis do pliku” (ze względu na potrzebę opracowania odpowiedniej formy i protokołu zapisu, jak również osiągnięcia wymaganych poziomów wydajnościowych). “Kreator nowych projektów” wymaga odpowiedniego zaprojektowania i implementacji potoku komunikacji z użytkownikiem i pobierania od niego niezbędnych informacji, jak również poprawnej i przemyślanej implementacji “Tworzenia nowego projektu”. Pozostałe elementy mogą być wspierane za pomocą istniejących już pod postacią bibliotek rozwiązań, co znacząco zmniejsza ich czasochłonność.

3.3. Podsumowanie

- *zakładana średnia prędkość sprintu* = 41 SP,
- *pojemność zespołu* = 112 h
- *oszacowanie godzinowe elementów Backlogu Sprintu* = 109 h < pojemność zespołu,
- ***szacowana prędkość*** = zakładana średnia prędkość sprintu/oszacowanie godzinowe elementów Backlogu Sprintu = **ok. 0,38 SP/h.**



4. Cel sprintu

Umożliwienie użytkownikowi zwizualizowania nowego projektu opartego na wnioskowaniu Mamdaniego oraz Sugeno w podstawowych kształtach.

5. Backlog sprintu

Sprint Numer 1							
ID Elementu	Podsumowanie	Story Points	Priorytet	Osoby Przydzielone	Szacunkowa Ilość Godzin	Status	Pozostała ilość godzin
#12	Tworzenie nowego projektu -Możliwość w interfejsie graficznym na wejściu programu opcji nowego projektu -Stworzenie przykładowego pustego projektu w systemie -Możliwość w interfejsie graficznym istniejącego projektu na utworzenie nowego -Zastąpienie aktualnego projektu pustym projektem	3	MUST	Adam Adam Adam Adam Adam	17h 2h 6h 2h 7h	Przed Rozpoczęciem	17h
#20	Wnioskowanie Mamdaniego i Sugeno -Możliwość wyboru wnioskowania podczas wejścia programu -Obsługiwanie wnioskowania Mamdaniego -Obsługiwanie wnioskowania Sugeno -Możliwość przekonwertowania istniejącego projektu z jednego wnioskowania na drugie	8	MUST	Filip i Julian Filip Filip Julian Julian	27h 1h 11h 11h 4h	Przed Rozpoczęciem	27h
#22	Definiowanie zmiennych i reguł rozmytych -Definiowanie zmiennych -Definiowanie reguł rozmytych	8	MUST	Filip Filip Filip	6h 2h 4h	Przed Rozpoczęciem	6h
#19	Wybór spośród 4 typów wykresów: dzwonowego, gaussowskiego, trójkątnego, trapezowego -Możliwość w interfejsie graficznym na wejściu programu wyboru między typami wykresów -Wykres dzwonowy -Wykres gaussowski -Wykres trapezowy -Wykres trójkątny -Możliwość zmiany typu wykresu w aktualnym projekcie	2	MUST	Filip, Adam i Julian Filip Filip Julian Adam Julian Filip	9h 0.5h 2h 2h 2h 2h 0.5h	Przed Rozpoczęciem	9h
#33	Rozszerzanie/zwężanie okienek -Możliwość zmiany wielkości okna przez złapanie za jego krawędź	2	SHOULD	Adam Adam	0.5h 0.5h	Przed Rozpoczęciem	0.5h
#21	Kreator nowych projektów -Możliwość w interfejsie graficznym na wejściu programu wejścia w kreator projektu -Definiowanie zmiennych wejściowych -Definiowanie wejściowych reguł rozmytych -Opcjonalna możliwość zdefiniowania dodatkowych zmiennych oraz reguł -Wybór pomiędzy wnioskowaniem Mamdaniego oraz Sugeno -Wybór typu wykresu	13	MUST	Julian i Adam Adam Julian Julian Julian Adam Adam	19.5h 2.5h 4h 4h 5h 2h 2h	Przed Rozpoczęciem	19.5h
#17	Szybki zapis do pliku -Możliwość szybkiego zapisu poprzez ikonę interfejsu graficznego -Możliwość szybkiego zapisu za pomocą skrótu klawiszowego -Wybór domyślnej lokalizacji zapisu -Wybór pomiędzy .txt, .csv oraz .bin jako domyślnego formatu szybkiego zapisu -Konwersja danych programu do formatu .txt -Konwersja danych programu do formatu .bin -Konwersja danych programu do formatu .csv -Przeprowadzenie zapisu do pliku o domyślnym rozszerzeniu, w domyślnej lokalizacji	5	SHOULD	Filip, Adam i Julian Adam Adam Filip Filip Julian Julian Filip Adam	30h 1.5h 0.5h 1h 1h 8h 8h 8h 2h	Przed Rozpoczęciem	30h



6. Kryteria akceptacji

6.1. Tworzenie nowego projektu

Tworzenie nowego projektu #28

Open IkeaSzark/RPI Public

ReadySetGet opened 2 weeks ago · edited by Julkul1 Edits ▾ ...

Użytkownik może utworzyć nowy projekt poprzez interfejs użytkownika.

Nowy projekt jest tworzony w pustym stanie, bez żadnych wcześniej zdefiniowanych zmiennych, reguł rozmytych ani danych. Projekt zaczyna się z domyślnym układem roboczym.

Po utworzeniu projektu:

- Automatycznie wyświetla się okno robocze do edycji zmiennych rozmytych i reguł,
- Dane projektu są zapisywane w domyślnej lokalizacji, z możliwością późniejszego zapisania pod wybraną nazwą.

Projekt posiada nazwę domyślną (np. „Projekt 1”), którą użytkownik może zmienić przed zapisaniem lub na etapie zapisu do pliku.

Użytkownik może rozpocząć definiowanie zmiennych i reguł w nowym projekcie bez konieczności dodatkowego konfigurowania ustawień.

Opcje edycji projektu:

- Użytkownik może w każdej chwili zmienić nazwę projektu i zapisać go pod inną nazwą.
- Użytkownik ma możliwość otwarcia projektu bezpośrednio po jego utworzeniu.

Po utworzeniu projektu, użytkownik jest przekierowywany do ekranu głównego aplikacji, w którym może rozpocząć dalszą konfigurację systemu rozmytego.

Create sub-issue



6.2. Wnioskowanie Mamdaniego i Sugeno po kroku

Wnioskowanie Mamdaniego i Sugeno #20

Open IkeaSzark/RPI Public

ReadySetGet opened 2 weeks ago · edited by Julkul1 Edits ▾ ...

Implementacja mechanizmu wnioskowania Mamdaniego i Sugeno

- System obsługuje dwie metody wnioskowania: **Mamdaniego** (średnia ważona) i **Sugeno** (funkcje liniowe lub stałe).
- **Wydajność:** Czas obliczeń dla obu metod nie przekracza 1 sekundy dla 50 reguł.

Możliwość przełączania między metodami

- Użytkownik może łatwo przełączać metody bez opóźnień.
- **Wydajność:** Czas przełączania nie przekracza 200 milisekund.

Wyniki zgodne z teoretycznymi modelami

- Wyniki wnioskowania są zgodne z teoretycznymi przewidywaniami.
- **Wydajność:** Błąd wyników nie przekracza 0.01% w porównaniu do teoretycznych.

Obsługa różnych funkcji przynależności i defuzyfikacji

- Obsługuje funkcje trójkątne, trapezoidalne, gaussowskie, sigmoidalne oraz metody defuzyfikacji (np. centroid).
- **Wydajność:** Czas obliczeń nie przekracza 10 milisekund na funkcję.

Create sub-issue



6.3. Definiowanie zmiennych i reguł rozmytych

Definiowanie zmiennych i reguł rozmytych #22

Open IkeaSzark/RPI Public

ReadySetGet opened 2 weeks ago · edited by Julkul1 Edits

Dodawanie zmiennych rozmytych:

- Użytkownik może dodać dowolną liczbę zmiennych wejściowych i wyjściowych poprzez interfejs graficzny.
- Każda zmienna musi mieć:
 - unikalną nazwę,
 - określony zakres wartości (np. od 0 do 100),

Tworzenie zbiorów rozmytych:

- System pozwala na definiowanie kształtu zbioru (trójkątny, trapezowy, Gaussowski, dzwonowy).
- Użytkownik może ręcznie ustawić parametry funkcji przynależności.
- Wartości funkcji przynależności są wizualizowane na wykresie, który aktualizuje się w czasie rzeczywistym.

Tworzenie reguł rozmytych:

- Interfejs pozwala na tworzenie reguł w formie:
`Jeśli [warunek1] i/lub [warunek2] to [wynik]`
np. `Jeśli temperatura jest wysoka i wilgotność jest niska to wentylator jest szybki.`

Walidacja reguł:

- System sprawdza, czy każda reguła odwołuje się do istniejących zmiennych i zbiorów.
- Błędne lub niepełne reguły są oznaczane i nie mogą być zapisane.

Edytowanie i usuwanie:

- Użytkownik może edytować lub usuwać istniejące zmienne i reguły w dowolnym momencie.
- Zmiany są natychmiastowo odzwierciedlane w modelu systemu rozmytego.

Create sub-issue



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

Wybór spośród 4 typów wykresów: dzwonowego, gaussowskiego, trójkątnego, trapezowego #19

Open IkeaSzark/RPI Public

ReadySetGet opened 2 weeks ago · edited by Julkul1 Edits ...

Użytkownik ma możliwość wyboru jednego z czterech typów funkcji przynależności:

- trójkątna,
- trapezowa,
- gaussowska,
- dzwonowa.

Wybrany typ funkcji może być przypisany do zbioru rozmytego przez graficzny interfejs użytkownika.

Każdy typ funkcji:

- Jest poprawnie rysowany na wykresie,
- Posiada odpowiednią liczbę parametrów możliwych do edycji (np. a, b, c dla trójkąta, σ , c dla Gaussa, itp.),
- Zmiana parametrów powoduje natychmiastową aktualizację wykresu (bez potrzeby restartu systemu).

Dane o typie funkcji i jej parametrach są zapisywane do pliku przy eksporcie systemu i poprawnie odczytywane przy imporcie.

Dla każdego typu funkcji dostępny jest podgląd liczbowy (parametry) oraz graficzny (kształt na wykresie).

Create sub-issue



6.5. Rozszerzanie/zwężanie okienek

Rozszerzanie/zwężanie okienek #33

Open IkeaSzark/RPI Public

ReadySetGet opened 2 weeks ago · edited by Julkul1 Edits ▾ ...

Aplikacja umożliwia dynamiczną zmianę rozmiaru głównego okna oraz wszystkich głównych paneli/interfejsów użytkownika bez utraty funkcjonalności.

Wszystkie komponenty GUI automatycznie dostosowują swój układ i skalę do nowego rozmiaru okna.

Minimalny rozmiar okna jest ustalony, aby uniknąć nakładania się elementów lub ich obciążenia.

Działanie funkcji zostało przetestowane w zakresie:

- Maksymalizacji i minimalizacji okna,
- Ręcznego rozciągania (rogiem i krawędziami),
- Zmiany rozdzielczości ekranu (w systemie operacyjnym).

Wykresy i wizualizacje skalują się proporcjonalnie, zachowując czytelność osi i oznaczeń.

Nie występują błędy graficzne ani zacięcia przy zmianie rozmiaru — GUI zachowuje pełną funkcjonalność.

Create sub-issue ▾



6.6. Kreator nowych projektów

Kreator nowych projektów #21

Open IkeaSzark/RPI Public

ReadySetGet opened 2 weeks ago · edited by Julkul1 Edits ▾ ...

Kreator prowadzi użytkownika przez minimum 3 etapy konfiguracyjne:

- Nazwa projektu i opcjonalny opis,
- Definicja zmiennych wejściowych i wyjściowych (liczba, nazwy, zakresy),
- Wybór domyślnego typu funkcji przynależności dla każdej zmiennej (np. trójkątna, trapezowa itp.),
- Wybór pomiędzy wnioskowaniem Sugeno oraz Mamdaniego.

W każdym etapie:

- Wszystkie pola wymagane są jasno oznaczone,
- Walidacja danych następuje natychmiast.

Użytkownik może:

- Przechodzić między krokami „Dalej” / „Wstecz” bez utraty danych,
- Anulować kreator w dowolnym momencie — z potwierdzeniem odrzucenia wprowadzonych danych.

Po zakończeniu kreatora:

- Projekt jest tworzony automatycznie z wprowadzonymi danymi,
- Użytkownik jest przekierowany do środowiska głównego, gdzie może kontynuować edycję projektu,
- System potwierdza utworzenie projektu poprzez komunikat końcowy.

Create sub-issue



6.7. Szybki zapis do pliku

Szybki zapis do pliku #17

Open IkeaSzark/RPI Public

ReadySetGet opened 2 weeks ago · edited by Julkul1 Edits ▾ ...

Użytkownik ma dostęp do opcji „Szybki zapis” w menu, pasku narzędzi lub skrótem klawiszowym, Ctrl+S, która natychmiast zapisuje projekt do ostatnio używanej ścieżki pliku.

Jeśli projekt nie był wcześniej zapisany:

- System automatycznie otwiera okno dialogowe zapisu z domyślną nazwą pliku i lokalizacją,
- Po pierwszym zapisaniu, kolejne szybkie zapisy nie wymagają potwierżeń — dane są nadpisywane automatycznie.

Zawartość pliku obejmuje:

- Wszystkie zdefiniowane zmienne rozmyte i ich funkcje przynależności,
- Reguły rozmyte,
- Ustawienia projektu.

System wyświetla krótkie powiadomienie o powodzeniu operacji.

W przypadku błędu zapisu (np. brak dostępu do pliku, błędna ścieżka):

- Użytkownik otrzymuje czytelną informację o problemie i możliwość zapisania pliku do innej lokalizacji.

Format pliku:

- Może być tekstowy (np. .json, .xml, .fis, .txt) — zawartość da się odczytać i edytować ręcznie,
- Struktura zapisu zgodna z wcześniejszymi projektami, umożliwiającą łatwy odczyt przez system po ponownym otwarciu.

Wydajność:

- Czas zapisu nie przekracza 1 sekundy dla projektów o standardowej złożoności (do 20 zmiennych i 50 reguł).

Create sub-issue



7. Definicja ukończenia (DoD)

- Wszystkie testy jednostkowe i integracyjne zakończyły się sukcesem,
- Funkcjonalność została zaprezentowana i zaakceptowana przez interesariuszy,
- System działa zgodnie z wymaganiami i nie zawiera błędów krytycznych,
- Dokumentacja została zaktualizowana zgodnie z nową funkcjonalnością,
- Wymagania niefunkcjonalne zostały przetestowane,
- Nowa funkcjonalność nie powoduje regresji,
- W przypadku funkcji związanych z uczeniem (ANFIS, PyTorch), działanie zostało sprawdzone na co najmniej jednym kompletnym przepływie danych (end-to-end).

8. Bibliografia

- platforma do przeprowadzenia spotkania Planning Poker: <https://planningpokeronline.com/>,
- materiały wykładowe dr inż. Jakuba Milera.