

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ**

**ЦЕНТР ЗАОЧНОЇ, ДИСТАНЦІЙНОЇ ТА ВЕЧІРНЬОЇ ФОРМ НАВЧАННЯ**

**КАФЕДРА КОМП'ЮТЕРНИХ НАУК**

## **ВИПУСКНА РОБОТА**

**на тему:**

**«Ергономічна оцінка інтерфейсу»**

**Завідувач  
випускаючої кафедри**

**Довбиш А. С.**

**Керівник роботи**

**Барченко Н. Л.**

**Студент групи ІНз-71с**

**Дубовіков І. О.**

**Суми 2021**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
Центр заочної, дистанційної і вечірньої форм навчання  
Кафедра комп'ютерних наук

Затверджую \_\_\_\_\_

Зав. кафедри Довбиш А. С.

“ \_\_\_\_\_ ” \_\_\_\_\_ 2021 р.

**ЗАВДАННЯ**

**до випускної роботи**

Студента четвертого курсу, групи ІНз-71С спеціальності “Комп’ютерні науки” заочної форми навчання Дубовікова Ігоря Олександровича.

**Тема: “Ергономічна оцінка інтерфейсу”**

Затверджена наказом по СумДУ

№ \_\_\_\_\_ от \_\_\_\_\_ 2021 р.

**Зміст пояснювальної записки:** 1) аналітичний огляд актуальності проблеми; 2) Дослідження методів вирішення проблеми; 3) опис основних положень, математичних моделей і критеріїв, що використовуються для оцінки ергономічності інтерфейсу; 5) розробка інформаційного й програмного забезпечення інтелектуальної системи; 6) аналіз результатів моделювання.

Дата видачі завдання “ \_\_\_\_\_ ” \_\_\_\_\_ 2021 г.

Керівник випускної роботи \_\_\_\_\_ Барченко Н. Л..

Завдання прийняв до виконання \_\_\_\_\_ Дубовіков І. О.

## **Реферат**

**Записка:** 28 стор., 13 рис., 2 табл., 1 додаток, 13 джерел.

**Об'єкт дослідження** – оцінка ергономічної якості програмного інтерфейсу.

**Мета роботи** – розробка інформаційної системи для визначення ергономічної якості програмного забезпечення.

**Методи дослідження** – апарат нечіткої логіки.

**Результати** – розроблено алгоритм оцінки ергономічної якості програмних продуктів. Запропоновано використовувати систему нечіткого логічного виведення для оцінки ергономічності.

**СИСТЕМИ НЕЧІТКОГО ЛОГІЧНОГО ВИВЕДЕННЯ, ЕРГОНОМІЧНА  
ОЦІНКА, СИСТЕМА МАМДАНІ**

## ЗМІСТ

<b>ВСТУП</b> .....	5
<b>1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ЕРГОНОМІЧНОЇ ОЦІНКИ ПРОГРАМНОГО ІНТЕРФЕЙСУ</b> .....	6
1.1 Актуальність ергономічної оцінки.....	6
1.2 Показники ергономічності.....	7
<b>2 МЕТОДИ ДОСЛІДЖЕННЯ ЕРГОНОМІЧНОГО ОЦІНЮВАННЯ ПРОГРАМНИХ ІНТЕРФЕЙСІВ</b> .....	11
2.1 Система нечіткого логічного виведення.....	11
2.2 Середовище розробки MATLAB.....	13
<b>3 РОЗРОБКА ІНФОРМАЦІЙНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ</b> .....	15
3.1 Постановка задачі.....	15
3.2 Програмна реалізація.....	18
<b>ВИСНОВКИ</b> .....	26
<b>СПИСОК ЛІТЕРАТУРИ</b> .....	27
<b>ДОДАТОК А. Структура FIS</b> .....	28

## ВСТУП

Метою дипломної роботи є розробка системи оцінювання ергономічності програмного продукту на основі нечіткої логіки для визначення ефективності застосування програми.

З кожним днем все більше послуг, починаючи від сфери освіти і до медицини, переходять в онлайн. Зростає попит, зростає пропозиція, і конкуренція серед онлайн-майданчиків також прагне вгору. Грамотне і зручне оформлення систем необхідно не тільки щоб залучити користувачів, але і щоб їх утримати.

Однією з головних завдань ергономіки є забезпечення зручності використання людиною продуктів, що виробляються і систем. Її рішення припускає адаптацію проектів продуктів або систем, включаючи дисплеї, характеристики приладів, програмні засоби, робочі місця, виробниче середовище та робочі завдання до характеристик, можливостям і обмеженням потенційних користувачів. Поліпшення ергономічних властивостей систем підвищить якість роботи, зменшить помилки і дискомфорт, мінімізує ризики, пов'язані зі здоров'ям та безпекою. Показати, як це відбувається, можна за допомогою теорії нечіткої логіки.

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ЕРГОНОМІЧНОЇ ОЦІНКИ ПРОГРАМНОГО ІНТЕРФЕЙСУ

## 1.1 Актуальність ергономічної оцінки

Розвиток технологій все більше ускладнює принципи побудови та використання систем. Зараз нам багато чого може здаватися буденним, але історично прості і вже звичні нам завдання раніше зовсім не були такими легкими у виконанні.

Наприклад, робота в текстовому редакторі або взаємодія з веб-сторінками. Легкість і доступність роботи з цими речами - в чималому ступені заслуга не тільки технічного прогресу, а й оптимізації інтерфейсів. Фахівці адаптують інтерфейси під кінцевого користувача - в тому числі і перш за все підтем, який не володіє спеціальними знаннями.

Як наслідок, це полегшує і прискорює навчання, розширює список доступних навичок, спрощує вирішення низки завдань, або дозволяє вирішувати абсолютно нові, підвищує продуктивність праці.

З точки зору сприйняття значення ергономічності наступне:

- Користувачі можуть сконцентруватися на виконанні своїх завдань в нормальному робочому процесі. Їм не потрібно розбиратися в меню інтерфейсу або розуміти внутрішню архітектуру програмного забезпечення, структуру елементів управління, значення іконок і кнопок. Іншими словами, їм не потрібно думати, як конвертувати завдання в команду «введення» для комп'ютера і як зробити повний цикл операцій.
- Користувачам не потрібно досконально розуміти принципи функціонування комп'ютерного обладнання та програмного забезпечення.

- Його користувачі не можуть бути відвернуті або спантеличені інтерфейсом в процесі взаємодії з ним. Всі етапи роботи і конкретні операції прості, зрозумілі і відтворювані з очікуваним результатом.
- Користувачі можуть нормально працювати з інтерфейсом в більшості умов і сценаріїв.

З точки зору користувача, він може спокійно і продуктивно працювати з програмою і не відчувати себе некомпетентним.

З точки зору розробника, юзабіліті - важливий аспект затребуваності і популярності системи в цілому, адже вона створюється для кого-то.

З точки зору менеджера, продукти з поганою ергономікою зажадають більше часу і енергії на освоєння від користувача, а це дуже погане конкурентну «перевага».

Нарешті, з точки зору участі інтерфейсів в продажах - інтернет-магазини, каси і термінали самообслуговування - людина не зможе купити те, що він не може знайти.

## **1.2 Показники ергономічності**

Проведений аналіз великого числа використовуваних показників і даних показав, що для оцінки програмних продуктів можна використовувати п'ять показників - швидкість виконання операцій користувачами (далі - швидкість роботи), кількість скоєних помилок, швидкість навчання користувачів, суб'єктивну задоволеність користувачів і візуальну привабливість для користувача інтерфейсу (або інакше - показники технічної естетики). Вимірювання п'яти даних показників дозволяє отримати комплексну оцінку зручності використання.



Рисунок 1.1- показники ергономічності

Розглянемо зміст кожного оцінюваного параметра і спосіб його обробки.

### 1. Швидкість роботи користувачів

Оцінка швидкості виконання типових дій користувача може проводитися за допомогою експертної оцінки. Найбільш поширений метод експертної оцінки, що спирається на методи моделювання дій користувачів (GOMS, CPM, KLM, NGOMSL, CMN-GOMS і т. і.). Їх суть полягає в тому, що виділяються типові дії при роботі з додатком, будуються алгоритми цих дій і оцінюється середній час виконання кожного алгоритму. В процесі оцінки кожен алгоритм розбивається на ряд простих операцій (натискання на кнопку, переклад курсору на екрані, аналіз подальших дій і т. і.), Для яких існують тимчасові значення, отримані на великій кількості користувачів. Для програмних продуктів можна виділити три основні аспекти:

- Швидкість обробки даних
- Інтуїтивна зрозумілість і зручність навігації
- Зручність роботи зі змістом;

### 2. Кількість скоєних помилок

Кількість помилок оцінюється за допомогою юзабіліті-тестування, проведеного на групі користувачів. В процесі тестування користувачі повинні виконати ті ж дії, які використовувалися для оцінки часу роботи типових дій експертами. Виконання дій на екрані монітора фіксуються на відео і, після проведення тестування, будуються алгоритми виконання всіх дій. Помилкою вважається невиконання дії і відхилення в алгоритмах дій у користувача і у



експерта. Результатом вимірювання є кількість помилок по всім діям в цілому і по кожній дії окремо. Отримані помилки групуються за типами (моторні помилки, друкарські помилки, не розуміння логіки роботи програми і т.і.) або результатами їх виникнення (критичні, некритичні).

### 3. Швидкість навчання користувачів

Швидкість навчання пов'язана з наявністю якісних засобів навчання і цілісної, несуперечливої інформаційною моделлю додатки. Для засобів навчання оцінюється якість і повнота програмної документації (опис типових дій, налаштувань, що виникають помилок і т. і.), наявність засобів навчання в інтерфейсі (пошук, контекстна довідка) і додаткових коштів підтримки (спеціалізовані форуми, навчальні матеріали на офіційних сайтах компаній і т.і.). Під інформаційною моделлю розуміється структура програмного забезпечення, яка містить інформацію про його стан і функціонування. При оцінці інформаційної моделі аналізується такі показники, як:

- Наявність структурування призначеного для користувача інтерфейсу, налаштувань, документації і звітів програми.
- Наявність в інтерфейсі в кожен момент часу інформації про результати дій користувача, реакції програми на дії користувача і стан додатки.
- Наявність симетричності в програмних елементах, що виконують однотипні функції (наприклад, кнопки прийняття рішення або кнопки для переходу між програмними вікнами).
- Відсутність дублювання функцій, налаштувань, програмних вікон і елементів управління в різних компонентах програми.
- Відсутність нефункціональних програмних вікон, «битих» посилань і некоректною термінології.

### 4. Суб'єктивна задоволеність

Суб'єктивна задоволеність оцінюється на основі анкетування користувачів після роботи з додатком. Анкета спрямована на виявлення того,

наскільки користувачам було зручно працювати з додатком. В анкету входять як загальні питання ( «Чи було легко працювати з програмою?»), Так і локальні, що виявляють зручність роботи з різними інструментами програми - антивірусними компонентами, довідкою, звітами, настройками і т. і. Також користувачам пред'являлися відкриті питання ( «Опишіть у вільній формі виникають в процесі роботи з програмою незручності», «Які пропозиції щодо поліпшення програми Ви б могли запропонувати»).

## 5. Технічна естетика

Технічна естетика оцінюється комбінованим способом - за допомогою експертної оцінки та анкетування користувачів після роботи з додатком. Даний показник складається з оцінки використовуваних в інтерфейсі шрифтів, квітів, анімації, звукових сигналів, піктограм, програмних елементів, угруповань програмних елементів. Експертна оцінка дозволяє виявити помилки, пов'язані з порушенням правил проектування візуального та звукового оформлення призначеного для користувача інтерфейсу, а анкетування користувачів дозволяє отримати інформацію про привабливість реалізованих рішень, їх зрозумілості та зручності використання. Критерії для експертної оцінки показників технічної естетики спираються на вимоги державних стандартів з ергономіки в частині проектування та оцінки користувальницьких інтерфейсів. Наприклад, оцінка читаності шрифтів експертами може проводитися на відповідність вимогам щодо розміру знаків по ГОСТ Р ІСО 9355-2-2009 на поширених дозволах екрану. А при анкетуванні користувачів оцінка технічної естетики проводиться за допомогою питань «Всі написи в інтерфейсі легко читалися?», «Чи були зрозумілі використовувані в програмі піктограми?» і т.і.

## 2 МЕТОДИ ДОСЛІДЖЕННЯ ЕРГОНОМІЧНОГО ОЦІНЮВАННЯ ПРОГРАМНИХ ІНТЕРФЕЙСІВ

### 2.1 Система нечіткого логічного виведення

MATLAB дозволяє виконувати проектування систем нечіткого логічного виведення за допомогою Fuzzy Logic Toolbox. Він має можливість створювати системи за допомогою зручного графічного інтерфейсу.

Базовим поняттям Fuzzy Logic Toolbox є FIS-структура – система нечіткого виведення (Fuzzy Inference System). FIS-структура містить усі необхідні дані для реалізації функціонального відображення “входи-виходи” на основі нечіткого логічного виведення відповідно до схеми, приведеної на рисунку 2.1

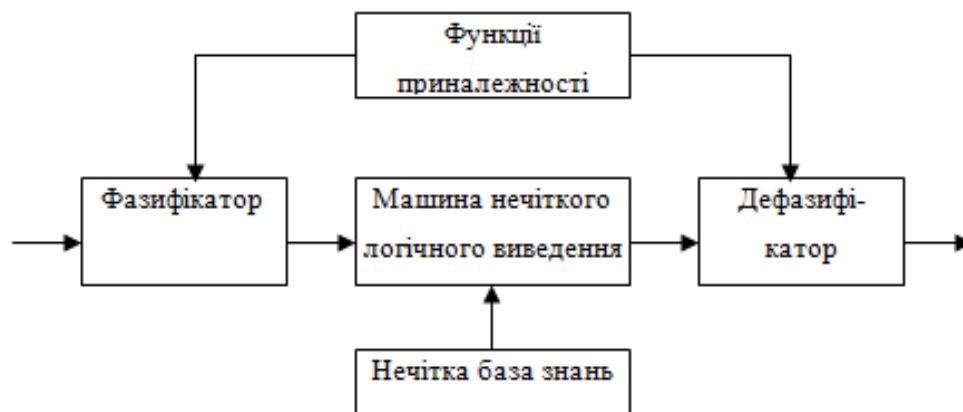


Рисунок 2.1- Система нечіткого виведення

Фазифікація (fuzzification) – це визначення ступеня виконання антецедентів правил. За допомогою фазифікації чіткому значенню ставляться у відповідність ступені його приналежності до нечітких множин .

Дефазифікація (defuzzification) – процедура перетворення нечіткої множини в чітке число за ступенем приналежності .

Фазифікатор перетворює чисельні вхідні значення в ступінь відповідності лінгвістичним змінним .

Нечітка база знань складається з блоків:

- база правил, що містить набір нечітких правил типу якщо-то;

– база даних, у якій визначені функції приналежності нечітких множин, що застосовуються в нечітких правилах.

Блок нечіткого логічного виведення виконує операції виведення на основі бази правил.

Блок дефазифікації перетворює результати в чіткі чисельні значення.

Система нечіткого логічного виведення представляється в робочій області MatLab у вигляді структури даних (рис.2.2).

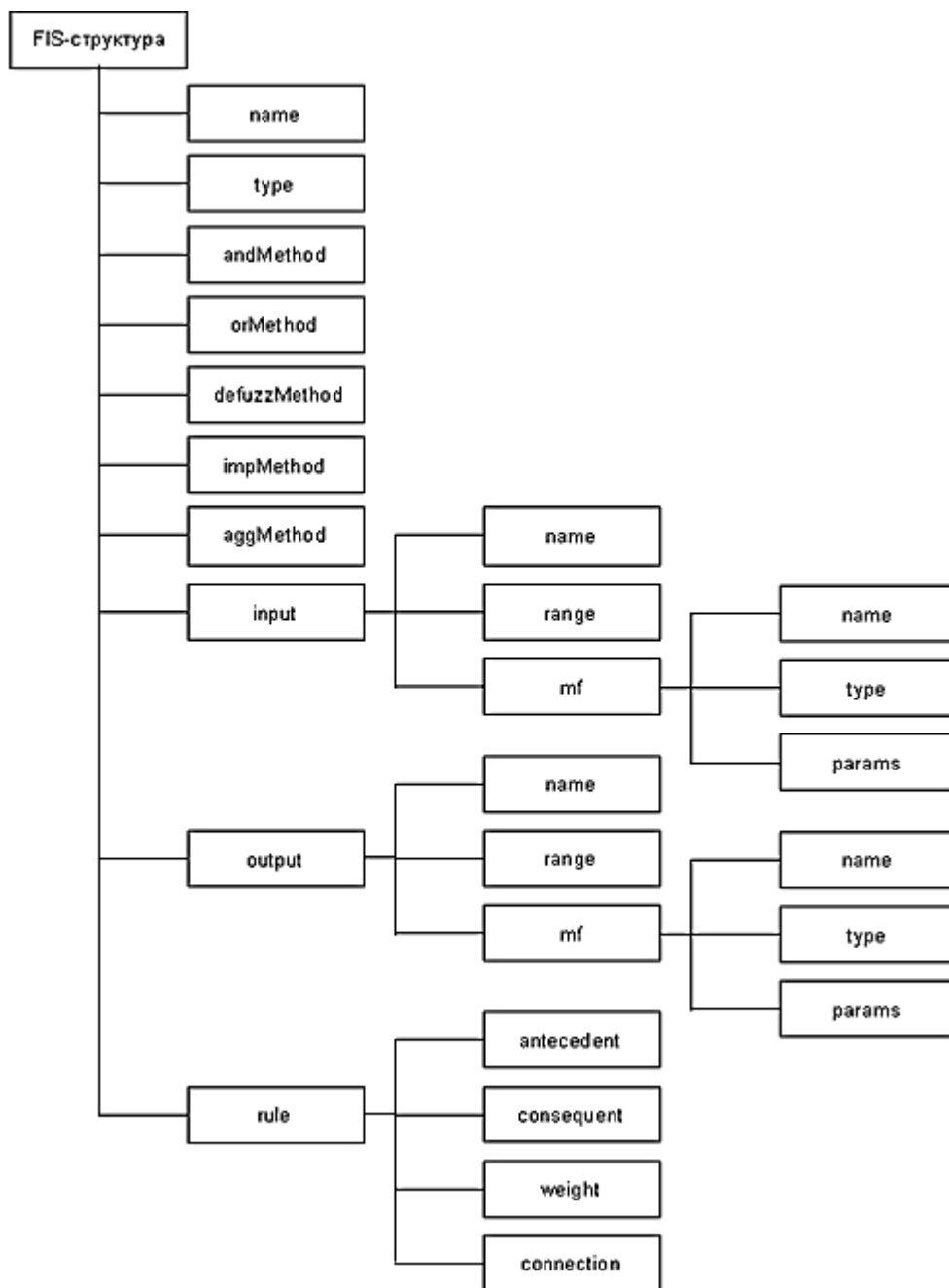


Рисунок 2.2 - FIS структура в Matlab

В пакеті Fuzzy Logic Toolbox реалізовані нечіткі моделі Мамдані та Сугено. Основна відмінність між ними полягає в способі задання значень вихідної змінної в правилах, що становлять базу знань.

В системах типу Мамдані значення вхідних змінних задаються нечіткими термами, в системах типу Сугено – як лінійна комбінація вхідних змінних.

Щоб відобразити взаємодію цих чинників між собою, треба створити нечіткі бази знань типу «Mamdani».

## 2.2 Середовище розробки MATLAB

Почнемо не зі стандартного екскурсу в історію і обговорення плюсів і мінусів мови, а з програмної середовища MATLAB. Просто уявіть собі графічний редактор, в якому ви зможете реалізувати будь-яку свою задумку, не маючи за плечима декількох років досвіду і відповідної освіти. І створивши один раз схему взаємодії інструментів, отримати якісний скрипт для багаторазового використання.

MATLAB - саме такий редактор в світі даних. Область його застосування безмежно широка: IoT, фінанси, медицина, космос, автоматика, робототехніка, безпроводні системи та багато-багато іншого. Загалом майже необмежені можливості по збору і візуалізації даних, а також прогнозування.

MATLAB - інструмент, що забезпечує взаємодію оператора (часто навіть не програміста) з усіма доступними можливостями аналізу, збору і представлення даних. У нього є очевидні плюси і мінуси, властиві мові живе в замкнутій екосистемі.

Недоліки:

- Повільний і перевантажений операторами, командами, функціями мови, основною метою якого є поліпшення візуального сприйняття.
- Вузькоспрямований. Немає ніякої більше програмної платформи, де б MATLAB був корисний.

- Дороге ПЗ. Якщо ви не студент - або готуйтеся спустошити кишені або перейти межу закону. І навіть якщо студент - ціна пристойна.
- Невисокий попит. Незважаючи на великий інтерес до MATLAB практично у всіх сферах, фактично і легально його використовують лише деякі.

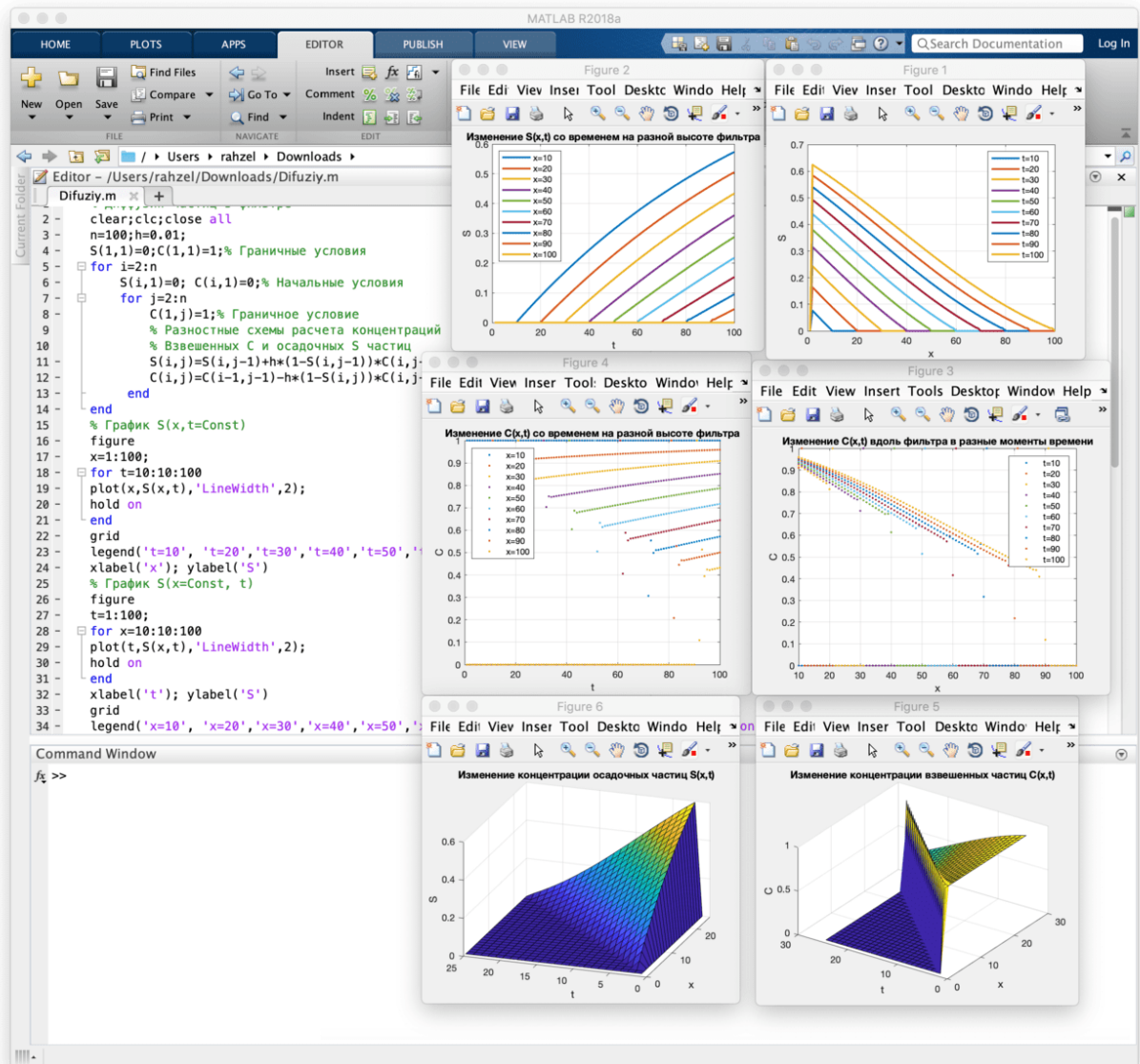


Рисунок 2.1 - робоче середовище Matlab

Переваги:

- Мова легка для вивчення, має простий і зрозумілий синтаксис.
- Величезні можливості. Але це скоріше перевага всього продукту в цілому.
- Часті оновлення, як правило помітні позитивні перетворення відбуваються не рідше як пару раз на рік.

- Програмне середовище дозволяє перетворювати його в "швидкий" код на C, C ++.

## 3 РОЗРОБКА ІНФОРМАЦІЙНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

### 3.1 Постановка задачі

Нехай задано деякий безліч локальних показників ергономічного якості модулів  $K = \{kj\}$ ,  $j = (1, n)$ . Окремі показники даної множини можуть бути виділені до деяких групи  $G = \{gi\}$ ,  $(1, m)$ .

Існує деяка процедура оцінювання відповідності локальних показників деяким вимогам. Задано безліч можливих результатів дизайн-ергономічної експертизи  $E = \{e1, e2, e3\}$ . Оцінка якості модуля використовується для прийняття одного з таких рішень:  $e1$  - модуль відповідає заявленим дизайн-ергономічним рекомендацій оформлення,  $e2$  - потрібна доробка,  $e3$  - не відповідає заявленим дизайн-ергономічним вимогам оформлення.

Безліч конкретних аналізованих параметрів в кожному конкретному випадку залежить від багатьох факторів. У даній роботі обмежимося характеристиками, використовуваними для  $EE$ .

Позначимо через  $E$  - інтегральний показник якості ергономічної оцінки. Для оцінки цього показника будемо використовувати наступну інформацію:

$X$  – швидкість роботи користувачів, який оцінюється з урахуванням таких приватних показників:  $x1$  – швидкість обробки даних,  $x2$  - інтуїтивна зрозумілість і зручність навігації (вид і розташування кнопок управління, ключові точки переходів і довжина шляху до них, можливість довільного і послідовного пересування по матеріалу і т.п.),  $x3$  - зручність роботи зі змістом;  $Y$  – кількість скоєних помилок.

$Z$  – швидкість навчання користувачів, який оцінюється з урахуванням таких приватних показників:  $z1$  - наявність структурування призначеного для користувача інтерфейсу, налаштувань, документації і звітів програми,  $z2$  -

наявність в інтерфейсі в кожен момент часу інформації про результати дій користувача, реакції програми на дії користувача і стан додатки, z3 - наявність симетричності в програмних елементах, що виконують однотипні функції, z4 - відсутність дублювання функцій, налаштувань, програмних вікон і елементів управління в різних компонентах програми, z5 - відсутність нефункціональних програмних вікон, «битих» посилань і некоректною термінології.

V - Суб'єктивна задоволеність.

M - Технічна естетика.

Задача оцінки полягає в тому, щоб поставити у відповідність модулю з відомими приватними показниками одне з рішень.

1. Експертна оцінка значень локальних показників.
2. Відсіювання варіантів, в яких хоча б один ергономічний показник має значення нижче деякого критично допустимого.
3. Визначення певного інтегрального показника ергономічної якості за сукупністю локальних показників.

Етапи 1 і 2 досить повно описані в ергономічній літературі. Завдання, відповідна етапу 3, відноситься до задачі класифікації, яка може бути вирішена з залученням великої кількості методів, що дозволяють працювати з експертними оцінками (метод аналізу ієрархій, нейронні мережі, нечітка логіка). У зв'язку з нечіткістю інформації, що міститься в оцінках експертів, в якості найбільш перспективного підходу до багатокритеріальної оцінювання визначимо метод нечіткого логічного висновку, запропонований в роботі.

Загальна схема рішення задачі ергономічної оцінки на рис. 3.1 і являє собою послідовність таких дій:

1. Оцінка модуля по виділеним показниками по шкалі термометра
2. Процедура нечіткого логічного висновку



3. Прийняття рішення про відповідність досягнутих показників якості загальним і локальним ергономічним вимогам і встановлення ергономічного рівня якості інтерфейсу.

Якщо прийнято рішення про відповідність вимогам, то програма додається в бібліотеку програм і може бути використана для подальших процедур. Інакше, видаються рекомендації з доопрацювання або обґрунтовується невідповідність програмного інтерфейсу.

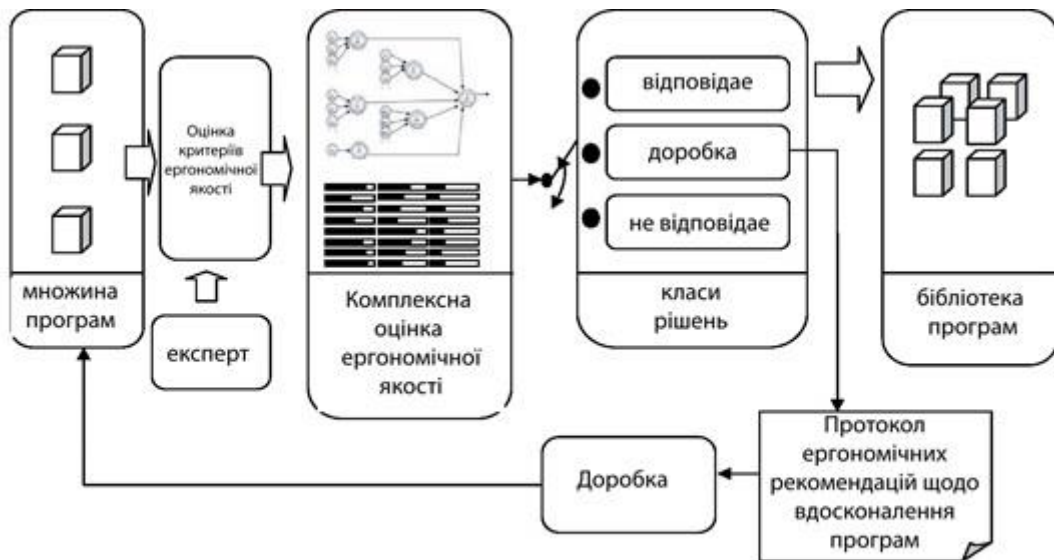


Рисунок 3.1 - Загальна схема рішення задачі ергономічної оцінки

Ієрархія показників відповідності показана на рис. 3.2 у вигляді дерева виводу, якому відповідає система співвідношень:

$$E = W_E(X, Y, Z, M, V)$$

$$X = W_X(x_1, x_2, x_3)$$

$$Y = W_{y(y_1)}$$

$$Z = W_Z(z_1, z_2, z_3, z_4, z_5)$$

$$M = W_{m(m_1)}$$

$$V = W_{v(v_1)}$$

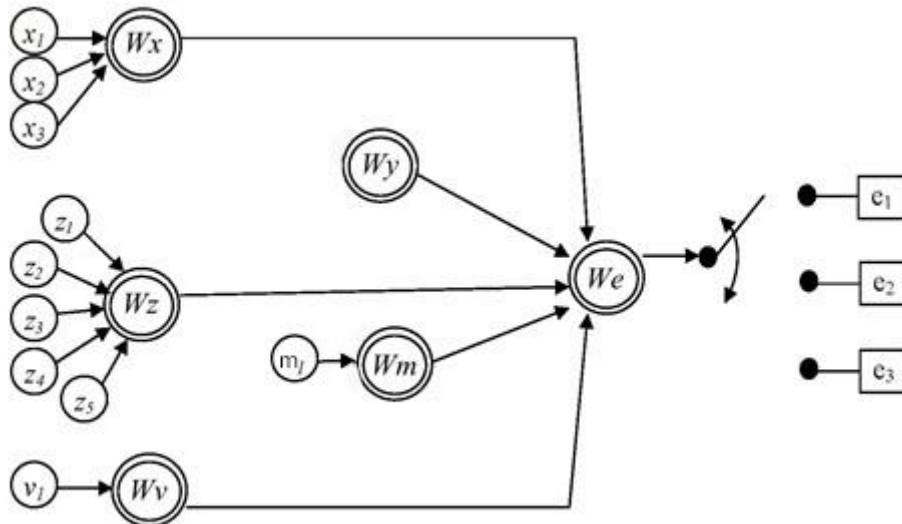


Рисунок 3.2 – Дерево виводу

### 3.2 Програмна реалізація

Всі отримані значення нормувалися і приводилися до діапазону від 0 до 100%. Залежно від типу вимірюваних параметрів обробка має свої особливості:

#### 1. Вимірювання часу типових операцій

Проводиться підсумовування мінімального часу виконання для кожної дії і приймається за час роботи з «ідеальним» продуктом (100%). Збільшення загального часу дій кожного продукту щодо часу ідеального продукту на 1 зменшує його підсумкову оцінку на 0,5%.

#### 2. Вимірювання кількості помилок

За 100% береться нульове кількість помилок. Після цього підсумовується кількість помилок за всіма діями на групі користувачів. Кожна критична помилка зменшує підсумкову оцінку на 1%, а некритична помилка - на 0,5%.

#### 3. Швидкість навчання

Окремо вважаються результати за двома анкетами - для оцінки навчальних матеріалів і для оцінки інформаційної моделі програми. Отримані значення усереднюються. Анкети обробляються наступним чином - за виконання кожного вимоги нараховується значення рівне  $100 / n$ , де  $n$  - це

кількість вимог в анкеті. За часткове виконання вимоги нараховується значення рівне  $100 / (2 * n)$ . Таким чином, оцінка по кожній анкеті буде нормована в діапазоні від 0 до 100.

#### 4. Суб'єктивна задоволеність

Після обробки анкети для кожного користувача виходять значення в діапазоні від 0 до 50 (10 питань по 5 градацій відповіді в кожному). Отримане значення для кожного користувача нормується, після чого вважається середнє значення оцінки для всієї групи користувачів, яке і є підсумковим значенням за даним показником.

#### 5. Технічна естетика

Окремо вважаються результати за двома анкетами - отримані при оцінці призначеного для користувача інтерфейсу користувачами та експертами. Анкета користувачів обробляється за аналогією з анкетою на суб'єктивну задоволеність, анкета експертів - по аналогії з анкетами на швидкість навчання. Отримані дані за анкетами для користувачів та експертів усереднюються. Після проведення обробки по кожному оцінюваному показником виходить 5 нормованих значень. Підсумковий показник розраховується в такий спосіб:

$$E = \sum A_i \times B_i$$

де E - значення ергономічності;

$A_i$  - вагові коефіцієнти для кожного оцінюваного показника;

$B_i$  - значення кожного оцінюваного показника.

Особливість приватних показників полягає в тому, що вони мають якісний характер, тобто не мають точного кількісного виміру. Тому при оцінці одного і того ж показника декількома експертами можуть виникати різні думки. Крім того експерт не завжди здатний словесно оцінити приватний показник, хоча інтуїтивно відчуває його рівень.

Для подолання цих труднощів можна оцінювати локальні показники за принципом термометра (рис. 3.3)



Рисунок 3.3 - Оцінка змінної  $u$  за принципом термометра

Зручність такого підходу полягає в тому, що різні за змістом локальні показники визначаються як лінгвістичні змінні, задані на єдиному універсальній множині  $U$   $u = [ , u ]$ , яким є шкала термометра. зниження суб'єктивізму може бути досягнуто за рахунок використання рекомендацій з табл. 3.1.

Наприклад, зафарбована частина шкали на рис. 3.3 для показника "Швидкість обробки даних", відповідає високій якості виконання.

Таблиця 3.1 - Деякі рекомендації по оцінці приватних показників якості

Приватний показник	Рівні оцінки показників			
	Недопустимий (Відкидається на попередньому етапі експертизи)	Мінімальний	Середній	Максимальний
швидкість обробки даних, $x_1$	Незручно працювати	Не зовсім комфортна робота	Є деякі зауваження	Комфортна робота
інтуїтивна зрозумілість і зручність навігації, $x_2$	Нестандартне розташування керуючих елементів	Потрібна попередня підготовка перед початком роботи з модулем	Є деякі зауваження	Інтуїтивно зрозумілий інтерфейс
зручність роботи зі змістом, $x_3$	Зміст відсутній	У змісті тільки основні розділи	Деякі незручності при переході	Зручний перехід між розділами і підрозділами

Припустимо, що лінгвістичні змінні  $x_1$ - $x_3$ ,  $y_1$ - $y_2$ ,  $z_1$ - $z_3$ ,  $v$ ,  $m_1$ - $m_3$  оцінюються нечіткими термами: Н - низький, С - середній, В - високий, які визначені за допомогою функцій належності.

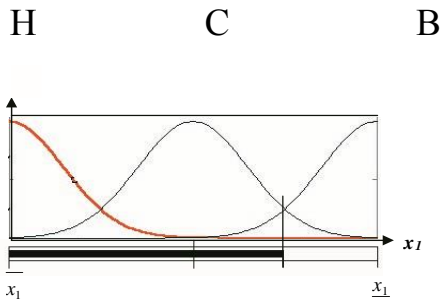


Рисунок 3.4 - Функції приналежності нечітких термів

Для моделювання укрупнених впливаючих чинників необхідно створити експертні нечіткі бази знань типа Мамдані, приклади яких наведені в табл.3.2. Елементи нечітких правил зв'язані логічною операцією ТА.

Таблиця 3.2 – Нечітка база знань для оцінки ергономічної якості інтерфейсу

Швидкість роботи	Кількість помилок	Швидкість навчання	Суб'єктивна задоволеність	Технічна естетика	Оцінка
Низька	Низька	Низька	Низька	Низька	Низька
Середня	Середня	Середня	Середня	Середня	Середня
Висока	Висока	Висока	Висока	Висока	Висока
Низька	Висока	Низька	Середня	Висока	Низька
Висока	Середня	Низька	Середня	Висока	Середня
Висока	Висока	Низька	Висока	Низька	Середня
Висока	Середня	Висока	Висока	Висока	Висока
Середня	Висока	Низька	Середня	Низька	Низька
Середня	Низька	Середня	Висока	Висока	Висока

Приклади графіків функцій належності нечітких термів “Низький” (Н), “Середній” (С) і “Високий” (В), реалізованих в Fuzzy Logic Toolbox в Matlab

представлені на рис.3.4-3.6. Графіки представлені як гаусівські функції на проміжку від -100 до 100. Отже, функція належності нечіткого терму «Низький» буде відповідати значенням, які лежать на проміжку від -100 до 50, а функція належності нечіткого терму «Високий» – від 50 до 100. Середнє значення терму «Середній» – у межах значення рівного 50.

Пояснимо сутність нечіткого терму «Високий» для кожного з впливаючих чинників (табл.3.2). Відповідно терм «Низький» буде дуже різнитися від «Високого» в гірший бік, а терм «Середній» буде виступати проміжним між термами «Високий» та «Низький».

Для прикладу візьмемо такі показники:

Швидкість роботи	Кількість помилок	Швидкість навчання	Суб'єктивна задоволеність	Технічна естетика	Оцінка
Низька	Висока	Низька	Середня	Висока	Низька

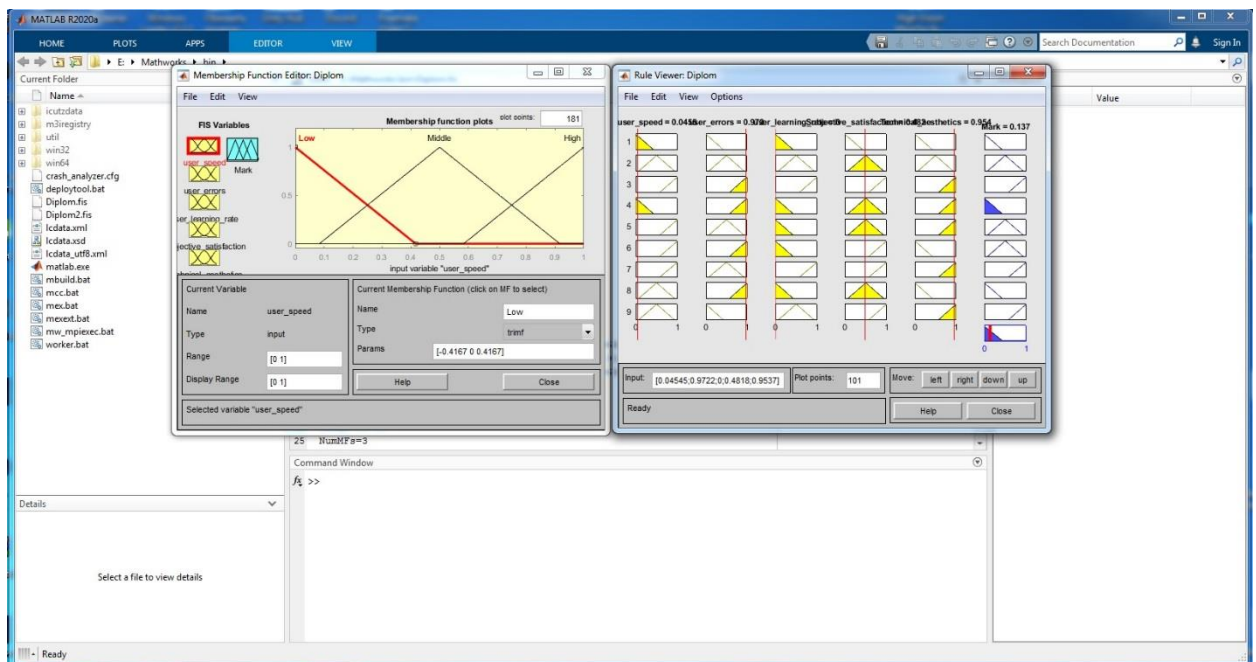


Рисунок 3.4 – «Низький» рівень оцінки

Для прикладу візьмемо такі показники:

Швидкість роботи	Кількість помилок	Швидкість навчання	Суб'єктивна задоволеність	Технічна естетика	Оцінка
Висока	Середня	Низька	Середня	Висока	Середня

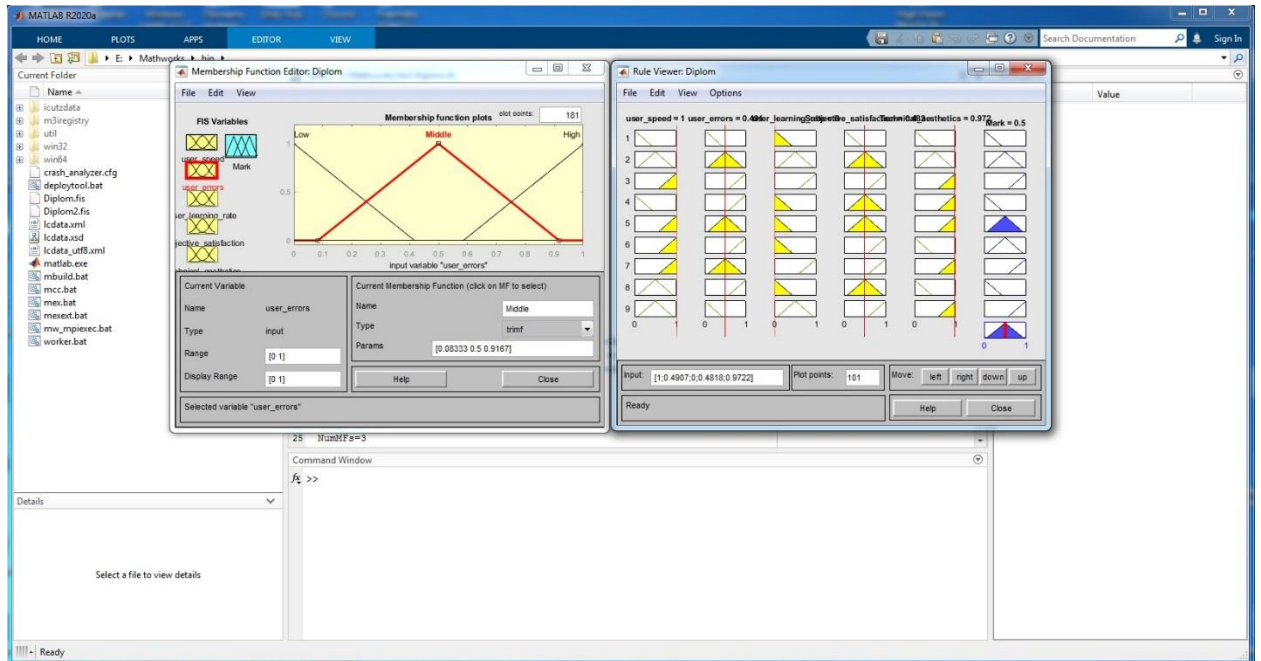


Рисунок 3.5 – «Середній» рівень оцінки

Для прикладу візьмемо такі показники:

Швидкість роботи	Кількість помилок	Швидкість навчання	Суб'єктивна задоволеність	Технічна естетика	Оцінка
Висока	Середня	Висока	Висока	Висока	Висока

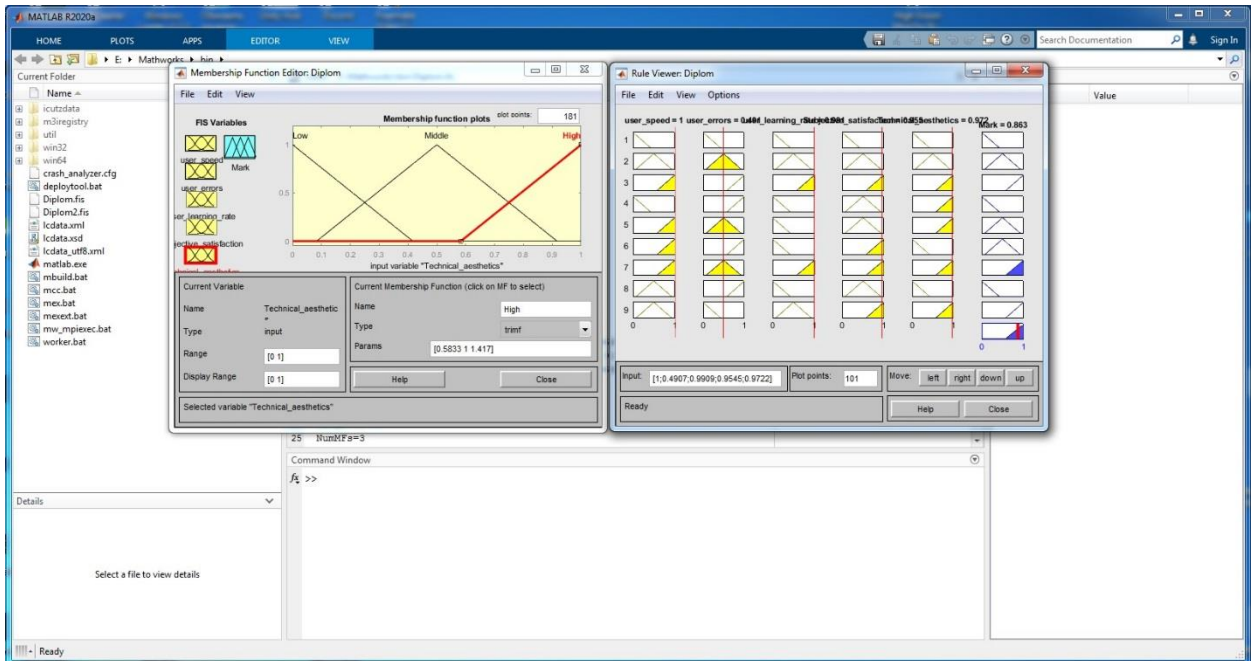


Рисунок 3.6 – «Високий» рівень оцінки

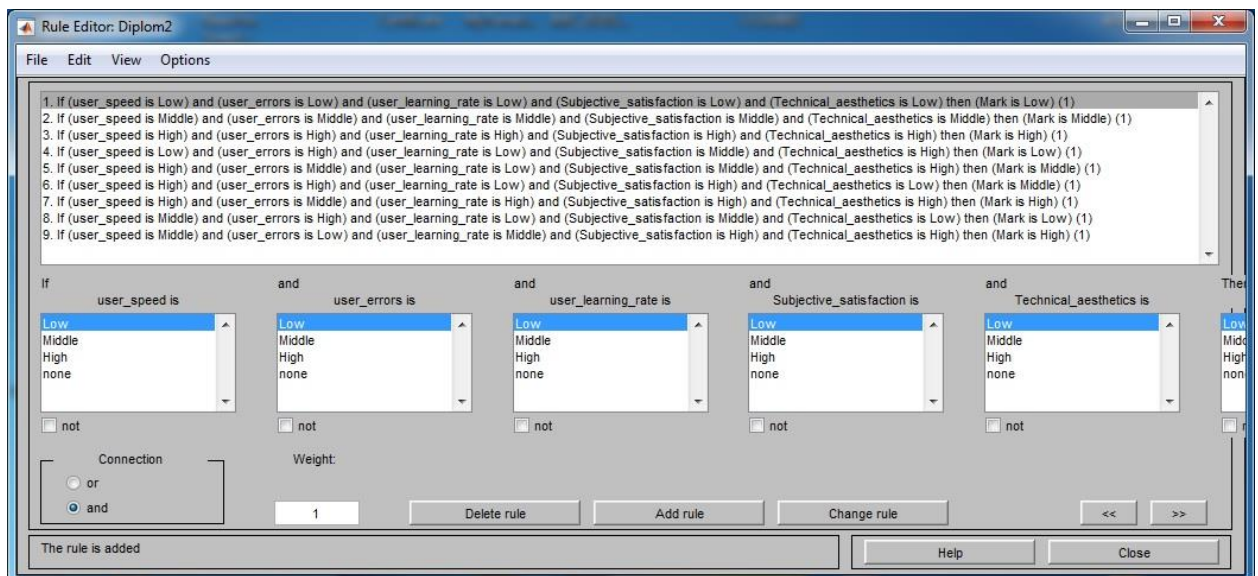


Рисунок 3.7 - Правила нечіткої бази знань для оцінки якості продукції

Правила нечіткої бази знань для оцінки якості продукції ми формуємо згідно таблиці 3.2.



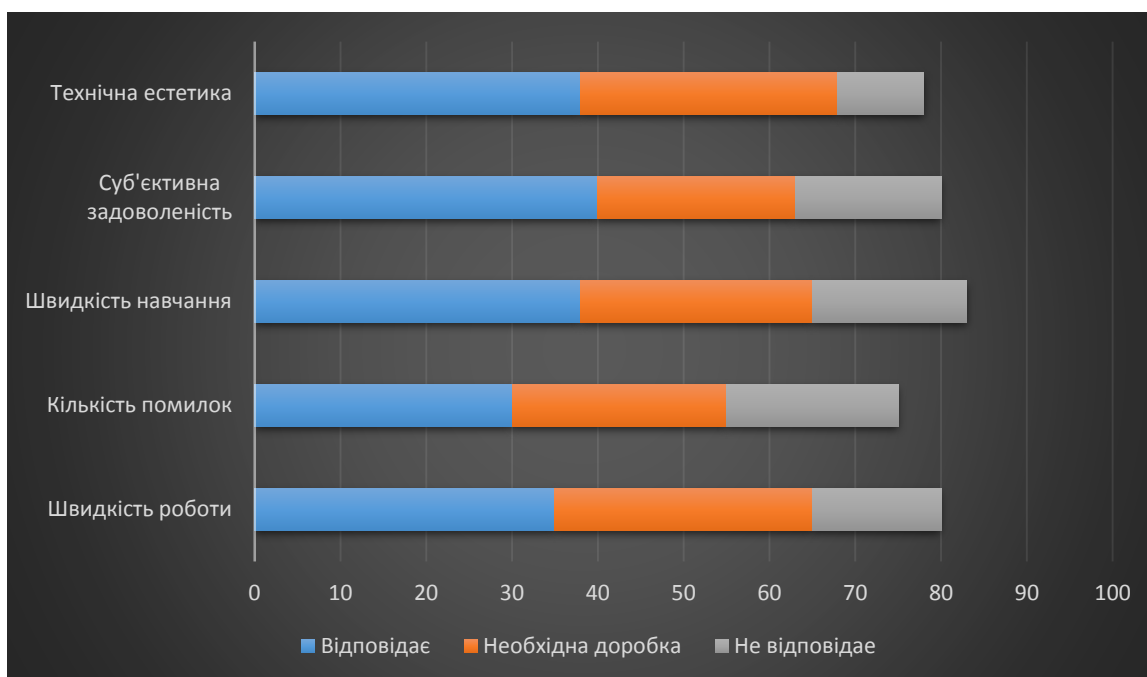


Рисунок 3.8 – Приклади оцінки інтерфейсів

Використовуючи наведений вище підхід формуємо множину програм, найбільш відповідних вимогам. Приклади оцінки трьох інтерфейсів представлені на рисунку 3.8.

## ВИСНОВКИ

В даній роботі розроблено інформаційну систему оцінки ергономічної якості програмних інтерфейсів. Сформований математичний опис системи оцінки ергономічної якості програмних інтерфейсів.

Для вирішення поставленого завдання застосована технологія нечіткого логічного виведення.

Програмну реалізацію виконано за допомогою Fuzzy logic ToolBox пакету Matlab.

Результати досліджень можуть бути використані для реалізації автоматизованого робочого місця експерта-ергономіста.

## СПИСОК ЛІТЕРАТУРИ

1. Зінченко В.П., Муніпов В.М. Основи ергономіки. - М.: Логос, 2001.
2. Головач В.В. Дизайн користувальницького інтерфейсу. 2001 - 141 с.
3. ГОСТ Р МЕК 60073-2000 Інтерфейс человекомашинная. Маркування та позначення органів управління і контрольних пристроїв. Правила кодування інформації
4. ДСТУ ISO 9355-1-2009 Ергономічні вимоги до проектування дисплеїв і механізмів управління. Частина 1. Взаємодії з людиною.
5. ДСТУ ISO 9355-2-2009 - Ергономічні вимоги до проектування дисплеїв і механізмів управління. Частина 2. Дисплеї
6. ДСТУ ISO 9241-11-2010 Ергономічні вимоги до проведення офісних робіт з використанням Дмитрій Мансуров (VDT). Частина 11. Настанови щодо забезпечення придатності використання
7. ДСТУ ISO 9241-110-2009 Ергономіка взаємодії людина-система. Частина 110. Принципи організації діалогу.
8. ДСТУ ISO 14915-1-2010 Ергономіка мультимедійних користувальницьких інтерфейсів. Частина 1. Принципи проектування та структура.
9. ДСТУ ISO 10075-2-2009 Ергономічні принципи забезпечення адекватності розумового навантаження. Частина 2. Принципи проектування.
10. Раскін Д., Інтерфейс: нові напрямки в проектуванні комп'ютерних систем. - Пер. з англ. - СПб: Символ-Плюс, 2004. – 272
11. Пономарьов І.А. Методи оцінки якості призначеного для користувача інтерфейсу.  
<http://itclaim.ru/Library/Books/ITS/wwwbook/ist6/ponomarev2/ponomarev2.html>
12. Мандел Т. Дизайн інтерфейсів. - М.: ДМК Пресс, 2005.
13. Kieras D. A Guide to GOMS Model Usability Evaluation using GOMSL and GLEAN3 - University of Michigan (<ftp.eecs.umich.edu/people/kieras>), 2002.

## ДОДАТОК А

```

[System]
Name='Diplom'
Type='mamdani'
Version=2.0
NumInputs=5
NumOutputs=1
NumRules=9
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='user_speed'
Range=[0 1]
NumMFs=3
MF1='Low': 'trimf', [-0.416666666666667 0 0.416666666666667]
MF2='Middle': 'trimf', [0.0833333333333333 0.5 0.916666666666667]
MF3='High': 'trimf', [0.583333333333333 1 1.416666666666667]

[Input2]
Name='user_errors'
Range=[0 1]
NumMFs=3
MF1='Low': 'trimf', [-0.416666666666667 0 0.416666666666667]
MF2='Middle': 'trimf', [0.0833333333333333 0.5 0.916666666666667]
MF3='High': 'trimf', [0.585978835978836 1.0026455026455 1.41931216931217]

[Input3]
Name='user_learning_rate'
Range=[0 1]
NumMFs=3
MF1='Low': 'trimf', [-0.416666666666667 0 0.416666666666667]
MF2='Middle': 'trimf', [0.0833333333333333 0.5 0.916666666666667]
MF3='High': 'trimf', [0.583333333333333 1 1.416666666666667]

[Input4]
Name='Subjective_satisfaction'
Range=[0 1]
NumMFs=3
MF1='Low': 'trimf', [-0.416666666666667 0 0.416666666666667]
MF2='Middle': 'trimf', [0.0859788359788361 0.502645502645503 0.91931216931217]
MF3='High': 'trimf', [0.583333333333333 1 1.416666666666667]

[Input5]
Name='Technical_aesthetics'
Range=[0 1]
NumMFs=3
MF1='Low': 'trimf', [-0.416666666666667 0 0.416666666666667]
MF2='Middle': 'trimf', [0.0833333333333333 0.5 0.916666666666667]
MF3='High': 'trimf', [0.583333333333333 1 1.416666666666667]

[Output1]
Name='Mark'
Range=[0 1]
NumMFs=3
MF1='Low': 'trimf', [-0.416666666666667 0 0.416666666666667]
MF2='Middle': 'trimf', [0.0833333333333333 0.5 0.916666666666667]
MF3='High': 'trimf', [0.583333333333333 1 1.416666666666667]

[Rules]
1 1 1 1 1, 1 (1) : 1
2 2 2 2 2, 2 (1) : 1
3 3 3 3 3, 3 (1) : 1
1 3 1 2 3, 1 (1) : 1
3 2 1 2 3, 2 (1) : 1
3 3 1 3 1, 2 (1) : 1
3 2 3 3 3, 3 (1) : 1
2 3 1 2 1, 1 (1) : 1
2 1 2 3 3, 3 (1) : 1

```