**РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРОЕКТУВАННЯ ЕКСПЕРТНИХ СИСТЕМ**

**DEVELOPMENT OF SOFTWARE FOR DESIGNING EXPERT SYSTEMS**

Науковий керівник: к.т.н., доцент кафедра інформаційних технологій

Рудніченко Микола Дмитрович

Здобувач бакалаврату Войцеховський Андрій Сергійович

Supervisor: (Ph.D), assistant professor Department of Information Technology

Rudnichenko Mykola Dmytrovych

Bachelor's candidate Wojciechowski Andriy Serhiyovych

**Анотація:** Мета роботи полягає в дослідженні особливостей процесів проектування та управління експертними системами, що побудовані на базі використання різних моделей уявлення знань шляхом розробки прикладного десктоп-орієнтованого програмного забезпечення. В рамках даної роботи розглянуто задачу проектування експертних систем для різних областей і сфер людської діяльності, розглянуто різні моделі подання знань.

**Annotation:** The aim of the work is to study the features of the processes of design and management of expert systems, built on the use of different models of knowledge representation through the development of desktop-oriented application software. In the framework of this work the problem of designing expert systems for different areas and spheres of human activity is considered, different models of knowledge representation are considered.

**Ключові слова:** експертні системи, програмне забезпечення, технології проектування.

**Key words:** expert systems, software, design technologies.

На початку вісімдесятих років минулого століття в дослідженнях зі штучного інтелекту сформувався самостійний напрямок, який одержав назву «експертні системи» (ЕС).

Метою досліджень в цьому новому напрямку була розробка програм, які при вирішенні завдань, складних для експерта-людини, отримують результати, що не поступаються за якістю і ефективністю рішень, що отримуються експертом [1-3].

Для позначення цієї дисципліни також часто використовують термін «інженерія знань», введений Е.Фейгенбаумом як «привнесення принципів і інструментарію досліджень з області штучного інтелекту до рішення важких прикладних проблем, що вимагають знань експертів». Основним призначенням ЕС є розробка програмних засобів, які при вирішенні завдань, важких для людини, одержують результати, які не поступаються за якістю і ефективністю вирішення, рішенням одержуваних людиною-експертом [4].

У зв'язку з цим актуальним завданням є проведення дослідження особливостей проектування і застосування ЕС для різних тематичних областей і моделей подання знань в ЕС, а також розробка прикладного програмного забезпечення організації процесів управління такими системами в інтерактивному режимі [5].

Проектована система містить 5 окремих програмних модулів (рис.1) с відповідною логікою побудови моделей уявлення знань:

1. Модуль побудови продукційних моделей.

2. Модуль побудови семантичних моделей.

3. Модуль побудови фреймових моделей.

4. Модуль експорту проектованої ЕС до окремого файлу (підримується формат \*.json).

5. Модуль симуляції процесу роботи створеної ЕС у режимі «Питання-відповідь».



Рисунок 1 – Структурна схема модулів програмного застосування

Після запуску системи та переходу до відповідної форми здійснюється створення нового шаблону ЕС засобами передбачених компонентів інтерфейсу системи.

Після завантаження шаблону (створено структури даних для зберігання та обробки правил, даних та знань) здійснюється створення нових питань, змінних та вихідних значень шляхом використання текстових віджетів для вводу даних.

На базі введених даних стає можливим додавання нових логічних правил для формування бази знань ЕС у структурному вигляді. У разі проведення успішної перевірки (верифікації) даних на логічну суперечливість (тобто дані введено без порушення логічного висновку) здійснюється збереження даних до програмних структур (колекцій) та вивід результатів у вигляді текстового повідомлення про успішне створення ЕС.

У зворотному випадку користувач буде проінформований про виникнення помилки у вигляді діалогового вікна та буде направлений до етапу введення.

Після створення ЕС можливим є перехід до інтерактивного тестування ЕС у режимі питань-відповідей.

Якщо на всі наведені системою запитання користувачем надано відповіді здійснюється вивід результатів логічного висновку, після чого користувач може здійснити експорт створеної ЕС до файлу формату \*.json та завершити роботу системи [6].

У разі, якщо хоча б одна з відповідей не була надана користувачем, то буде видано повідомлення про помилку вибору у вигляді відповідного діалогового вікна.

Користувач системи має можливості:

* створити нову ЕС (створення та видалення питань, відповідей, правил, оновити дані та завдати назву ЕС);
* створити модель семантичної мережі (завдати вхідні та вихідні дані, створити зв’язки між об’єктами, побудувати та експортувати граф семантичної моделі);
* створити фреймову модель;
* закрити активний проект ЕС;
* відкрити існуючу модель з файлу;
* зберегти створену модель до файлу;
* побудувати дерево рішень (у графічному вигляді в формі графу);
* вийти з системи.

Структура системи, яка розроблена засобами середовища розробки PyCharm, наведена нижче:

а) Backend, каталог, що містить класи, які відповідають за експорт та збереження даних, зокрема:

1. EmptyExpertEngine.py, клас, що реалізує логіку експорту шаблону розробленої ЕС до файлу;

2. Serializer.py, клас, що імплементує процедуру серіалізації даних до файлу з можливістю подальшого відкриття та використання в рамках створеної системи.

б) ESKnowledgeBase, каталог, що зберігає створені бази знань у форматі json за схемою «Питання»-«Значення»-«Логічні правила».

в) GUI, каталог, що містить класи та підкаталоги для розміщення відповідних класів обробки даних та допоміжних ресурсів для формування графічного інтерфейсу створюваної системи. До основних з них відносяться наступні каталоги та класи:

1. Images, каталог, що містить графічні ресурси для відображення у інтерфейсі користувача. Здебільшого це графічні файли формату \*.png та \*.jpg, які інтегруються до проекту та відображаються у відповідних формах та вікнах;

2. Models, каталог, що містить наступні програмні класи моделей обробки даних:

3. AnswersListModel.py, клас, що імплементує додавання, збереження та обробку даних по відповідях користувача під час використання проектованої ЕС;

4. ESThemesListModel.py, клас, що реалізує загальні можливості по обробці та візуальному відображенню теми та даних у графічних віджетах форм інтерфейсу;

5. QuestionsListModel.py, клас, що імплементує додавання, збереження та обробку даних по питаннях для користувача під час заповнення змісту проектованої ЕС;

6. RulesListModel.py, клас, що імплементує додавання, видалення, збереження та обробку даних по логічних правилах у продукційній моделі уявлення знань під час проектування змісту правил ЕС;

7. SemanticInputTableModel.py, клас, що реалізує логіку додавання вхідних даних до відповідного віджета таблиці для побудови моделі семантичної мережі з уявлення знань;

8. SemanticObjectListModel.py, клас, що забезпечує логіку створення, зміни, збереження та загальну обробку даних по побудові чи перебудові структури семантичної мережі;

9. SemanticOutputTableModel.py, клас, що реалізує логіку додавання вихідних даних до відповідного віджета таблиці для побудови моделі семантичної мережі з уявлення знань;

10. Views, каталог, що містить наступні програмні класи обробки та розмітки форм користувальницького інтерфейсу системи:

11. MainWindowView.py, клас, що забезпечує процес створення та відображення головної форми системи, а також реалізує обробку зв’язків між відповідним файлом розбиття та розміщення елементів графічного інтерфейсу MainWindow.ui, що створено засобами PyQtDesigner;

12. MainWindow.ui, файл розмітки головного вікна створюваної системи графічного інтерфейсу користувача.

Висновки. Практичне значення розробленого програмного забезпечення полягає у наданні користувачу можливостей зі створення експертних систем різного тематичного напрямку.

**Література:**

1. Астахова І. С. Системи штучного інтелекту / І. С. Ірпінь. - М.: Біном. Лабораторія знань, 2011. - 314 с.

2. Бєляков А. Г. Аналіз достовірності висновків, що формуються за допомогою експертно-статистичних систем / О. Г. Біляков, А. С. Мандель. - М.: Інститут проблем управління ім. В.А. Трапезнікова РАН, 2002. - 64 с.

3. Гаврілова Т. А. Основи знань інтелектуальних систем / Т.А. Гаврилова, В.Ф. Хорошівський. - СПб.: Пітер, 2000. - 384 с.

4. Рудниченко Н. Д. Прототип программного обеспечения проектирования экспертных систем / Н. Д. Рудниченко, В. Н. Манькевич // Інформатика, інформаційні системи та технології: тези доповідей п’ятнадцятої всеукраїнської конференції студентів і молодих науковців. Одеса, 27 квітня 2018р. - Одеса, 2018. – С. 71.

5. Гежа М. І. Аналіз особливостей та етапів виконання завдання прогнозування при аналізі даних / Гежа М. І., Тищенко С. Є., Рудніченко М, Д // XXI Всеукраїнська науково-технічна конференція молодих вчених, аспірантів та студентів .Одеса, 22-23 квітня 2021 р. - Одеса, 2021. - С.85-86.

6. Rudnichenko N. Decision Support System for the Machine Learning Methods Selection in Big Data Mining / N. Rudnichenko, V. Vychuzhanin, I. Petrov, D. Shibaev // Proceedings 0f The Third International Workshop on Computer Modeling and Intelligent Systems (CMIS-2020): session 6 “Intelligent Information Technologies” April 27-May 1, 2020. – Zaporizhzhia: NU “Zaporizhzhia Polytechnic” (edited by S. Subbotin), 2020. − P. 872-885.