Ukrainian Journal of Educational Studies
and Information Technology

Vol. 5. No. 3. September 2017

http://ojs.mdpu.org.ua/index.php/itse

УДК 004.891.2

Проектування довідкової інтелектуальної експертної системи для вівчарства у приватних господарствах країни

Дмитро Лубко

Таврійський державний агротехнологічний університет, м. Мелітополь, Україна,
кандидат технічних наук, доцент кафедри комп'ютерних наук,
orcid.org/0000-0002-2506-4145, di75ma@gmail.com

Анотація. Робота присвячена проектуванню довідкової інтелектуальної експертної системи (ДІЕС) для вівчарства. Описано хід її проектування та її функціональні можливості. Розроблена довідкова інтелектуальна експертна система дозволяє виводити рекомендації та поради на екран комп'ютера з вибору раціональної та ефективної технології розведення овець, а також друкувати отримані дані. Розроблена система має дворівневу структуру, а саме – діалоговий вибір вхідних критеріїв (даних), а також модуль логічної обробки та виведення раціональних рекомендацій щодо технології (вихідні дані). Розроблена система призначена у першу чергу для фермерів приватних господарств країни або зацікавлених приватних господарів. Зазначається, що використання даної системи дозволить приватним господарствам збільшити виробництво та якість м'яса, вовни, смушок та хутра при розведенні овець. А це у свою чергу дозволить зменшити витрати на технологію при їх розведенні, зберегти час фермеру, заощадити на ліках, кормах, що у свою чергу підвищить прибутки та рентабельність господарств.

Ключові слова: інтелектуальна експертна система; вівчарство; довідковий модуль; вівці; технологія розведення.

Design of a reference intelligence expert system for sheep breeding in national private farms

Dmytro Lubko

Tavria State Agrotechnological University, Melitopol, Ukraine,
Candidate of Engineering Sciences, Associate professor, Associate professor of the Department of Computer Science,
orcid.org/0000-0002-2506-4145, di75ma@gmail.com

Summary. The paper deals with the design of a reference intelligence expert system for sheep breeding. The process of its design and its functional capabilities are described. The developed reference intelligence expert system makes it possible to display recommendations and tips on the computer screen to select a rational and effective sheep breeding technology, as well as to print the received data. The developed system has a two-tier structure, namely, dialog selection of input criteria (data), as well as a module for logical processing and output of reasonable technology recommendations (output data). The step-by-step method of designing the system is

ISSN: 2521-1234 online

© The Author(s)

determined. The first step is to obtain a technical specification from the customer (farm, enterprise, farmer, etc.) for the development of a reference system. The second step is to determine criteria for the technology in keeping with the farm requirement specification according to the literature and sheep breeding requirements. The third step is to identify the most important factors influencing the process of sheep raising for each of these criteria. The fourth step is to define the main production rules for which the system will be programmed, namely, the module for logical knowledge processing for this technology (and these are the input parameters (factors) when designing the system). The fifth step is to determine the main output criteria (factors) that will be displayed after processing the input rules of the system based on logical deduction rules according to the appropriate sheep breeding technology. The sixth step is to design buttons for more convenient system usage, in addition, if necessary, or at the request of the customer (for example, a button for deleting previous information in windows, a button for storing recommendations in a separate text file, the exit button, etc.). The seventh step is to test the system by users and the customer. The eighth step is to adopt the developed system by the customer and correct it, if necessary. The ninth step is to provide maintenance of the developed system. The block of input data of the developed intelligence expert system has the following elements: a) sheep handling (pasture-stall, stall-pasture); b) sheep condition (higher condition, average condition, lower than average condition); c) sheep breeding type (meat, milk, meat and milk, wool); d) sheep feeding (meat-and-wool sheep, rams of meat-and-goat sheep, sucking females to lactation); e) methods of breeding (purebred, crossing). The block of output factors where appropriate rational recommendations for sheep breeding technology comprises: a) recommendations for sheep handling; b) recommendations for treatment of sheep diseases; c) recommendations for sheep feeding; d) sheep slaughter and storage of products; e) recommended sheep breeds; f) recommendations for sheep breeding; g) characteristics of products. The developed system is intended primarily for private national farmers or interested private householders. It is noted that the use of this system will allow private households to increase the production and quality of meat, wool, lambskin and furs when breeding sheep. This will reduce the cost of the breeding technology, save the farmer time, allow to save on medicines, feeds, which, in turn, will increase profits and profitability of farms.

Keywords: intelligence expert system; sheep breeding; reference module; sheep; breeding technology.

ВСТУП

Постановка проблеми. Вівчарство – це галузь тваринництва, що займається розведенням овець. Вівчарство забезпечує виготовлення сировини для легкої промисловості (вовна, овчина, смушки) і харчових продуктів (молоко, м'ясо, сало). Найціннішим продуктом є шерсть, яку використовують у виробництві тканин, трикотажу, килимів, валяних виробів, тощо. Зі шкір овець виготовляють хутряні вироби. З молока грубошерстих овець виробляють сири. Вівчарство відіграє також велику роль у постачанні продуктів харчування для населення. Важливим продуктом вівчарства є баранина. Хоч питома вага у м'ясному балансі її незначна, проте, маючи цінні поживні властивості, вона широко використовується як дієтичний продукт, виробництво якого значно дешевше інших видів дієтичного м'яса (Мирось, 2015). Вівчарство в Україні є традиційною галуззю та в економіці народного господарства відіграє визначну роль. Вівці постачають вовну для промисловості, необхідну для виробництва різних тканин, цінні каракульські смушки, що відіграють не останню роль у нашому експорті, дають молоко – важливий продукт харчування населення.

Враховуючи важливість та специфічність цієї галузі для населення та країни в цілому, дуже актуальним стає питання її активного розвитку та розширення поголів'я овець у країні. А це наразі можливо здійснити, тільки маючи необхідну інформацію з правильного розведення овець, маючи раціональні та ефективні технології з вирощування, годування, забою та переробки овець у певному приватному господарстві.

На сьогодні у багатьох приватних господарствах (ПГ) країни існує декілька (від 2 до 20 зазвичай) персональних комп'ютерів та своя локальна мережа. ПК та створена мережа потрібні для введення, обробки та систематизації зібраних даних та документації, а саме: готових контрактів, переліку технічних вимог до експлуатації різних видів техніки та устаткування, веденню бухгалтерії, готових звітів, тощо. Але цього для ефективного розвитку цих господарств та саме такого використання ПК – замало.

Ми вияснили, що для ефективного та швидкого розвитку приватних господарств у галузі вівчарства потрібно розробити для них відповідну інформаційну систему (програму, програмний модуль). Створення такої простої та сучасної системи (модуля) з технології вирощування овець, яка реалізовувала б та виводила раціональну (ефективну) технологію з вирощування овець у ПГ, і являє собою актуальну проблему, вирішенням якої ми і будемо займатися.

Тобто зі створенням такої системи для приватних господарств вони:

– будуть мати кращий, ефективніший та більш оперативний (швидкий) вибір технології з вирощування овець;
– будуть мати базу даних, в якій буде зберігатися інформація про всю технологію вирощування овець з можливістю її редагування, доповнення, видалення, тощо;
– будуть мати можливість зберігання отриманої технології (рекомендації) на ПК з можливістю прямого друку у файл або на принтер.

Все вищеназване у свою чергу надасть змогу якісно покращити роботу господарства, автоматизувати її систему накопичення даних, пришвидшити пошук насущної довідкової інформації про вирощування овець та в кінцевому етапі зможе принести ПГ значні прибутки. Це і визначає актуальність цієї роботи і власне статті.

При проектуванні даної системи ми вирішили, що доцільно розробити її у вигляді окремого (локального) модулю та створити її як довідкову систему.

Також було вирішено зробити її у вигляді інтелектуальної експертної системи, використовуючи відповідні рівні подання знань та продукційну модель. Сам модуль буде проектуватися за допомогою мови програмування C# у середовищі розробки Visual Studio.

Аналіз останніх досліджень. Питаннями розробки інтелектуальних, зокрема експертних, систем за допомогою різних інструментальних засобів займалося багато науковців та практиків (Шаров, 2016): А. П. Частіков (Clips), Д. І. Муромцев (Drools Guvnor), Б. Тейт, Е. Хювенен (Lisp), М. Брамер, С. П. Хабаров (Prolog), О. П. Солдатова (Visual Prolog) та інші вчені. Розробкою інтелектуальних систем у різних галузях

займалися Р. І. Баженов, М. С. Чванова, В. В. Осадчий, А. І. Леженко, М. Е. Желнін. Висвітленням питань щодо використання інтелектуальних інформаційних систем та засобів їх розробки займалися С. П. Хабаров, В. С. Тоіскін, Ю. Ф. Тельнов, В. В. Литвин, А. М. Козлов та інші дослідники.

Останні дослідження в тематиці розробки інтелектуальних експертних систем (ЕС) показали, що вони найчастіше використовуються як локальні системи (настільні) на підприємствах різної форми власності та розміру, у різноманітних державних установах (поліції, медицини, податкової, служби безпеки тощо). ДІЕС майже не розробляються для окремого приватного господарства. А це наразі дуже актуально, тому що при розробці спеціалізованої ДІЕС, наприклад, для окремого ПГ, будуть враховуватися специфічні та особливі технічні та інші дані самого цього господарства, що значно підвищить точність прогнозів та отриманих рекомендацій (аналог дорадництва). Саме тому для отримання великої ваги овець, якісного м'яса та хутра тощо працівникам галузі вівчарства треба завжди завчасно та кваліфіковано отримувати наукові знання з цієї предметної технології та кваліфіковану наукову підтримку спеціалістів. А це наразі можна буде зробити саме завдяки спроектованій системі (модулю).

Як показав проведений аналіз існуючих систем напрямку вівчарства – схожі аналоги є в досить малій кількості (знайдено два аналога). Основними їх недоліками є незначна кількість вхідних параметрів та невелика кількість параметрів для виведення, що говорить о низькій якості отриманих рекомендацій. Також у знайдених аналогів досить висока програмна вартість, що не всім господарям по кишені. У системі, яка нами розробляється, будуть відсутні вказані недоліки аналогів.

Метою даної роботи є розробка загальної методології та проектування довідкової інтелектуальної експертної системи для вівчарства у приватних господарствах країни з метою підвищення їх прибутків та рентабельності.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Різноманітні спеціалізовані довідкові інтелектуальні та експертні системи (ІЕС) знайшли своє призначення та використання у багатьох сферах нашого життя (сільському господарстві, тваринництві, науці, освіті, економіці тощо). Зазвичай вони призначені, у першу чергу, для генерування правильних та ефективних рішень у конкретній предметній галузі, отримання конкретних фахових рекомендацій у певній предметній області (наприклад, у вівчарстві). Як наслідок, попит на програмні продукти такого типу залишається на високому рівні. Однак для того, щоб створити якісну експертну систему, слід ретельно підійти до планування всіх етапів розробки, а також до вибору адекватного інструментального засобу, який дозволить розробити саме той програмний продукт, який задовольнить замовника (Шаров, 2016).

Особливістю сучасних діючих експертних систем є їх дуже вузька та предметна спеціалізація. Це системи для діагностики певного виду захворювань людини або для визначення структурної формули певного класу органічних з'єднань, або для пошуку оптимальної конфігурації конкретної обчислювальної системи. Вузька спеціалізація експериментальних експертних систем викликана бажанням зменшити об'єм професійних знань, що закладаються у систему, для спрощення задачі створення цих знань та їх збереження у пам'яті системи (Джарантано, & Гари, 2007; Нейлор, 2006).

Для представлення знань в експертних системах використовують різні способи. Найпоширеніші способи представлення знань використовують продукції. Продукція представляє порцію (квант) знання у формі правила типу «якщо..., то...».

Сама ЕС містить три типи знань (Уотермен, 2000; Джарантано, 2013):

а) структуровані знання про предметну ділянку – після того, як ці знання виявлені, вони не змінюються;

б) структуровані динамічні знання – змінні знання з предметної ділянки, які обновляються по мірі виявлення нової інформації;

в) робочі знання, які використовуються для розв'язування конкретної задачі або проведення консультації.

Всі перераховані знання зберігаються в базі знань. Для її побудови потрібно провести опитування спеціалістів, які є експертами у конкретній предметній ділянці, а потім систематизувати, організувати та індексувати отриману інформацію для простоти її використання (Таусенд, & Фохт, 2001).

Основу будь-якої інтелектуальної системи складають база знань і закладений у систему механізм виведення рішень. Ці компоненти визначають дві основні інтелектуальні характеристики системи: здатність зберігати знання про щось і вміння оперувати цими знаннями. Тому на початковому етапі розробки інтелектуальної системи нам потрібно визначитися із моделлю представлення знань.

Представлення знань в електронному варіанті передбачає визначення деяких початкових об'єктів, правил формування на їх основі нових об'єктів і у результаті отримання опису цих знань. Формальний спосіб опису предметної області і є моделлю представлення знань. У якості вхідних об'єктів виступають значення даних. Відносини між даними визначають правила утворення нових об'єктів. Виконуючи окремі процедури над відносинами між даними, структурують дані і формують певні знання.

У наш час моделі подання знань є предметом досліджень і розробок, як у середовищі вузьких спеціалістів (в основному програмістів і математиків), так і спеціалістів різноманітних предметних галузей, які намагаються використовувати інтелектуальні системи у своїй діяльності.

На сьогодні існують різноманітні моделі подання знань для різних предметних областей. Більшість з них може бути зведена до наступних класів: логічні моделі; семантичні мережі; фрейми; продукційні моделі та нейронні мережі. Звичайно, кожен із

зазначених вище типів моделей подання знань має свої переваги та недоліки у залежності від сфери застосування (Ревко, 2006; Козлов, 2013; Уинстон, 2009).

Для розробки нашої системи у якості моделі подання знань була обрана продукційна модель. *Продукційна модель* містить сукупність правил (продукції) у вигляді (Петров, 2003; Сахнюк, 2012):

1. «ЯКЩО» умова «ТО» дія;
2. «ЯКЩО» причина «ТО» наслідок;
3. «ЯКЩО» ситуація «ТО» рішення.

Суть моделі полягає у тому, що якщо виконуються певні правила умови, то потрібно зробити деяку дію. Продукційні моделі можуть бути реалізовані процедурно і декларативно. У процедурних системах неодмінно повинні бути: база даних, набір продукційних правил та інтерпретатор (він визначає послідовність активізації продукцій). База даних є змінною частиною моделі, а правила та інтерпретатор постійні. Можна додавати і змінювати лише факти (знання). Продукційні моделі застосовуються у тих предметних галузях, які характеризуються відсутністю чіткої логіки, а завдання вирішуються на основі незалежних правил (евристик).

На початку проектування нашої системи нами було вирішено провести візуальне моделювання в UML. UML можна представити як певний процес порівневого спуску від найбільш загальної і абстрактної концептуальної моделі вихідної системи до логічної, а потім і до фізичної моделі відповідної програмної системи. Для досягнення цих цілей спочатку будується модель у формі так званої діаграми варіантів використання (use case diagram), яка описує функціональне призначення системи або, іншими словами, те, що система буде робити у процесі свого функціонування. Діаграма варіантів використання є вихідним концептуальним уявленням або концептуальною моделлю системи у процесі її проектування і розробки.

Суть даної діаграми полягає у наступному (Фаулер, 2006): проектована система представляється у вигляді безлічі сутностей або акторів, що взаємодіють з системою за допомогою так званих варіантів використання. При цьому актором (actor) або дійовою особою називається будь-яка сутність, що взаємодіє з системою ззовні. Це може бути людина, технічний пристрій, програма або будь-яка інша система, яка може служити джерелом впливу на модельовану систему так, як визначить сам розробник. У свою чергу, варіант використання (use case) служить для опису сервісів, які система надає акторові. Іншими словами, кожен варіант використання визначає деякий набір дій, який чинять системою при діалозі з актором. При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів з системою.

Діаграма використання була розроблена за допомогою Case-засобу для створення UML-діаграм Diagram Designer (версії 1.27), який позиціонується як набір UML інструментів для бізнес аналізу, системного аналізу, що охоплює всі стадії розробки програмного забезпечення: аналіз, розробку, тестування і підтримку. Diagram Designer

також можна використовувати для формування системи вимог за умови використання мови UML для опису вимог.

Діаграма варіантів використання розробленої системи представлена на рис. 1. На діаграмі представлено три актори та десять прецедентів з варіантами відношень між собою.

Три актори (експерт, програміст, технологи) активно взаємодіють між собою під час проектування системи (рис. 1). Прецедент «ДІЕС» (сама оболонка програми) – система, під час роботи постійно взаємодіє з прецедентом «База знань ІЕС» з якої вона бере необхідні для роботи системи дані. Для початку роботи системи (ДІЕС) потрібно спочатку ввести вхідні дані (прецедент «Введення вхідних даних у ДІЕС»), далі система, використовуючи базу знань, компілює результат (прецедент «Виведення вихідних даних», тобто вже отримання вихідних рекомендацій). Сам прецедент «Виведення вихідних даних» має в собі та поділяється на сім інших прецедентів, які і являють собою рекомендації системи (ДІЕС) для технолога господарства (у нашому випадку з області вирощування овець), завдяки яким технолог будь-якого приватного господарства зможе проводити правильне та якісне вирощування овець (або для фермера ПГ).

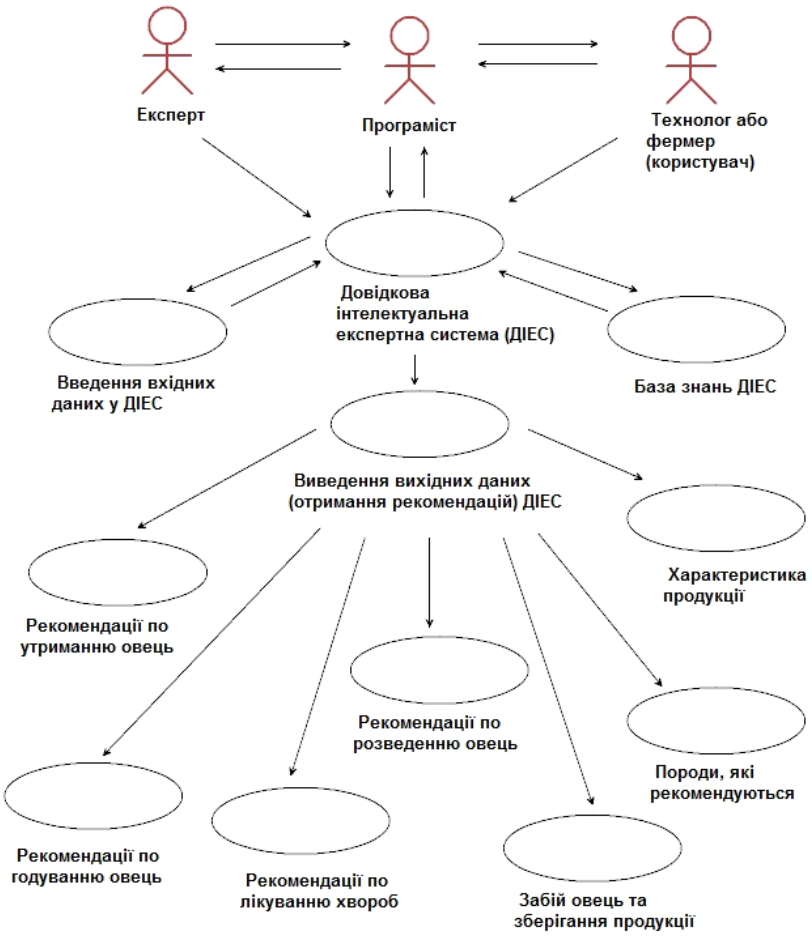

Рис. 1. Діаграма варіантів використання ДІЕС

Наступним етапом проектування системи була розробка функціональної моделі IDEF0.

IDEF0 – Function Modeling – це методологія функціонального моделювання і графічного опису процесів, призначена для формалізації і опису бізнес-процесів. Особливістю IDEF0 є її акцент на ієрархічне представлення об'єктів, що значно полегшує розуміння предметної області. Така модель є однією з найпрогресивніших моделей і використовується в організації бізнес-проектів і проектів, що базуються на моделюванні усіх процесів як адміністративних, так і організаційних (Черемных, Семенов & Ручкин, 2006).

Відмінною особливістю IDEF0 є її акцент на підпорядкованість об'єктів. В IDEF0 розглядаються логічні відносини між роботами, а не їх тимчасова послідовність (WorkFlow). Так само відображаються усі сигнали управління. Дана модель є однією з прогресивних моделей і використовується при організації бізнес-проектів та проектів, заснованих на моделюванні всіх процесів як адміністративних, так і організаційних.

Графічна мова IDEF0 дивно проста і гармонійна. В основі методології лежать чотири основні поняття (Черемных, Семенов & Ручкин, 2006), а саме:

1) Функціональний блок (Activity Box).

Функціональний блок графічно зображується у вигляді прямокутника і уособлює собою деяку конкретну функцію в рамках даної системи. По вимогам стандарту назва кожного функціонального блоку має бути сформульованою дієслівним способом (наприклад, «виробляти послуги», а не «виробництво послуг»).

Кожна з чотирьох сторін функціонального блоку має своє певне значення (роль), при цьому:

- верхня сторона має значення «Управління» (Control);
- ліва сторона має значення «Вхід» (Input);
- права сторона має значення «Вихід» (Output);
- нижня сторона має значення «Механізм» (Mechanism).

Кожен функціональний блок у рамках єдиної системи повинен мати свій унікальний ідентифікаційний номер.

2) Інтерфейсна дуга (Arrow).

Також інтерфейсні дуги часто називають потоками або стрілками. Інтерфейсна дуга відображає елемент системи, який обробляється функціональним блоком або надає інший вплив на функцію, відображену даним функціональним блоком.

Графічним відображенням інтерфейсної дуги є односпрямована стрілка. Кожна інтерфейсна дуга повинна мати своє унікальне найменування (Arrow Label). На вимогу стандарту найменування повинно бути оборотом іменника. За допомогою інтерфейсних дуг відображають різні об'єкти, в тій чи іншій мірі визначають процеси, що відбуваються

в системі. Такими об'єктами можуть бути елементи реального світу (деталі, вагони, співробітники і т.д.) або потоки даних і інформації (документи, дані, інструкції і т.д.).

Залежно від того, до якої з сторін підходить дана інтерфейсна дуга, вона носить назву «входить», «вихідної» або «керуючої». Крім того, «джерелом» (початком) і «приймачем» (кінцем) кожної функціональної дуги можуть бути тільки функціональні блоки, при цьому «джерелом» може бути тільки вихідна сторона блоку, а «приймачем» будь-яка з трьох, що залишилися.

Врахуйте, що будь-який функціональний блок за вимогами стандарту повинен мати принаймні одну керуючу інтерфейсну дугу і одну вихідну. Кожен процес повинен відбуватися за якимись правилами (що відображається керуючої дугою) і повинен видавати деякий результат (дуга, що виходить), інакше його розгляд не має ніякого сенсу.

Функціональна модель IDEFo розробленої системи наведена на рис. 2.

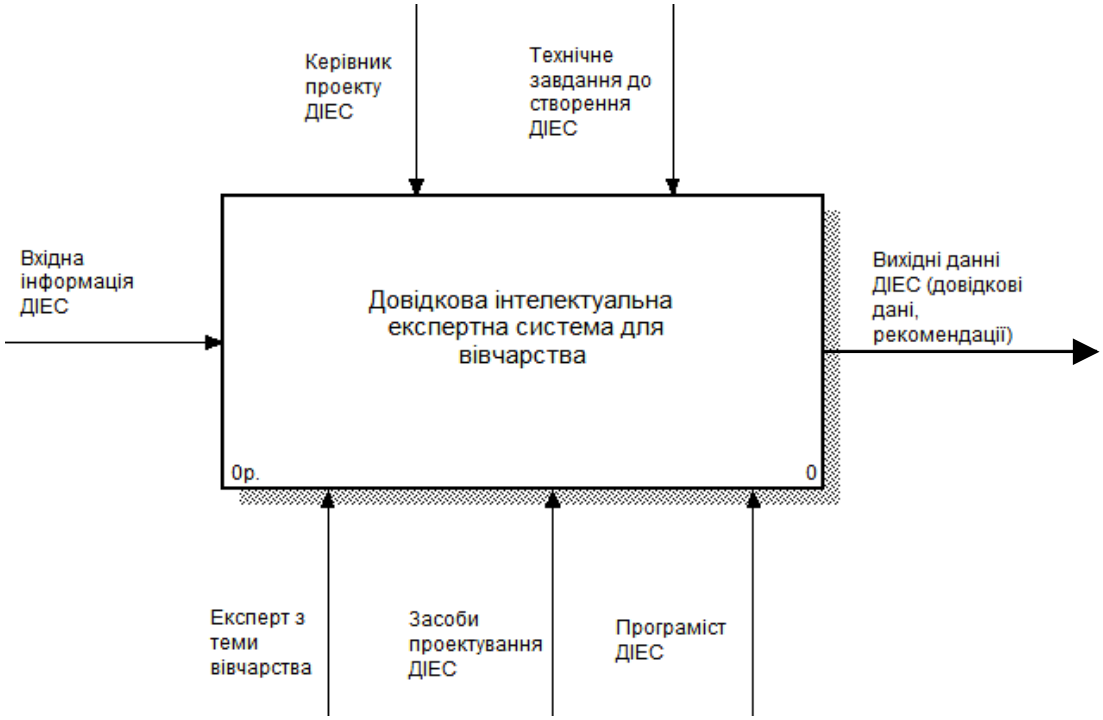

Рис. 2. Функціональна модель IDEFo розробленої ДІЕС

Після отримання функціональної моделі IDEFo розробленої системи потрібно зробити її декомпозицію (Decomposition).

Принцип декомпозиції застосовується при розбитті складного процесу на складові його функції (Черемных, Семенов & Ручкин, 2006). При цьому рівень деталізації процесу визначається безпосередньо розробником моделі. Декомпозиція дозволяє поступово і структуровано представляти модель системи у вигляді ієрархічної структури окремих діаграм, що робить її менш перевантаженою й легко засвоюються.

У процесі декомпозиції функціональний блок, який у контекстній діаграмі відображає систему як єдине ціле, піддається деталізації на іншій діаграмі. Отримана діаграма другого рівня містить функціональні блоки, що відображають головні підфункції функціонального блоку контекстної діаграми і називається дочірньою (Child diagram) по відношенню до нього (кожен з функціональних блоків, що належать дочірній діаграмі, відповідно називається дочірнім блоком – Child Box).

У свою чергу функціональний блок-предок називається батьківським блоком по відношенню до дочірньої діаграми (Parent Box), а діаграма, до якої він належить – батьківської діаграмою (Parent Diagram).

Кожна з підфункцій дочірньої діаграми може бути далі деталізована шляхом аналогічної декомпозиції відповідного їй функціонального блоку. Важливо відзначити, що в кожному разі декомпозиції функціонального блоку все інтерфейсні дуги, що входять у цей блок або виходять з нього, фіксуються на дочірній діаграмі. Цим досягається структурна цілісність IDEF0-моделі. Слід звернути увагу на взаємозв'язок нумерації функціональних блоків і діаграм – кожен блок має свій унікальний порядковий номер на діаграмі (цифра у правому нижньому куті прямокутника), а позначення під правим кутом вказує на номер дочірньої для цього блоку діаграми. Відсутність цього позначення говорить про те, що декомпозиції для даного блоку не існує.

Декомпозиція функціональних блоків розробленої системи наведена на рис. 3.

Проектування ДІЕС для вівчарства.

Цикл роботи експертної системи є по суті алгоритмом з логічним виведенням та обробкою даних. Логічне виведення може відбуватися багатьма способами, з яких найбільш поширені це – прямий порядок виведення і зворотний порядок виведення.

Для повноцінної роботи ДІЕС, яка розробляється, спочатку потрібно створити відповідну базу даних у яку буде заноситися вся предметна інформація (про овець), а потім розробити програмний інтерфейс системи (оболонку).

Визначимо загальну методологію розробки ДІЕС.

Спочатку виявимо та розглянемо основні критерії даної технології та виявимо деякі основні правила, які використовуються у ДІЕС, для визначення її логічного апарату та процесу виведення даних.

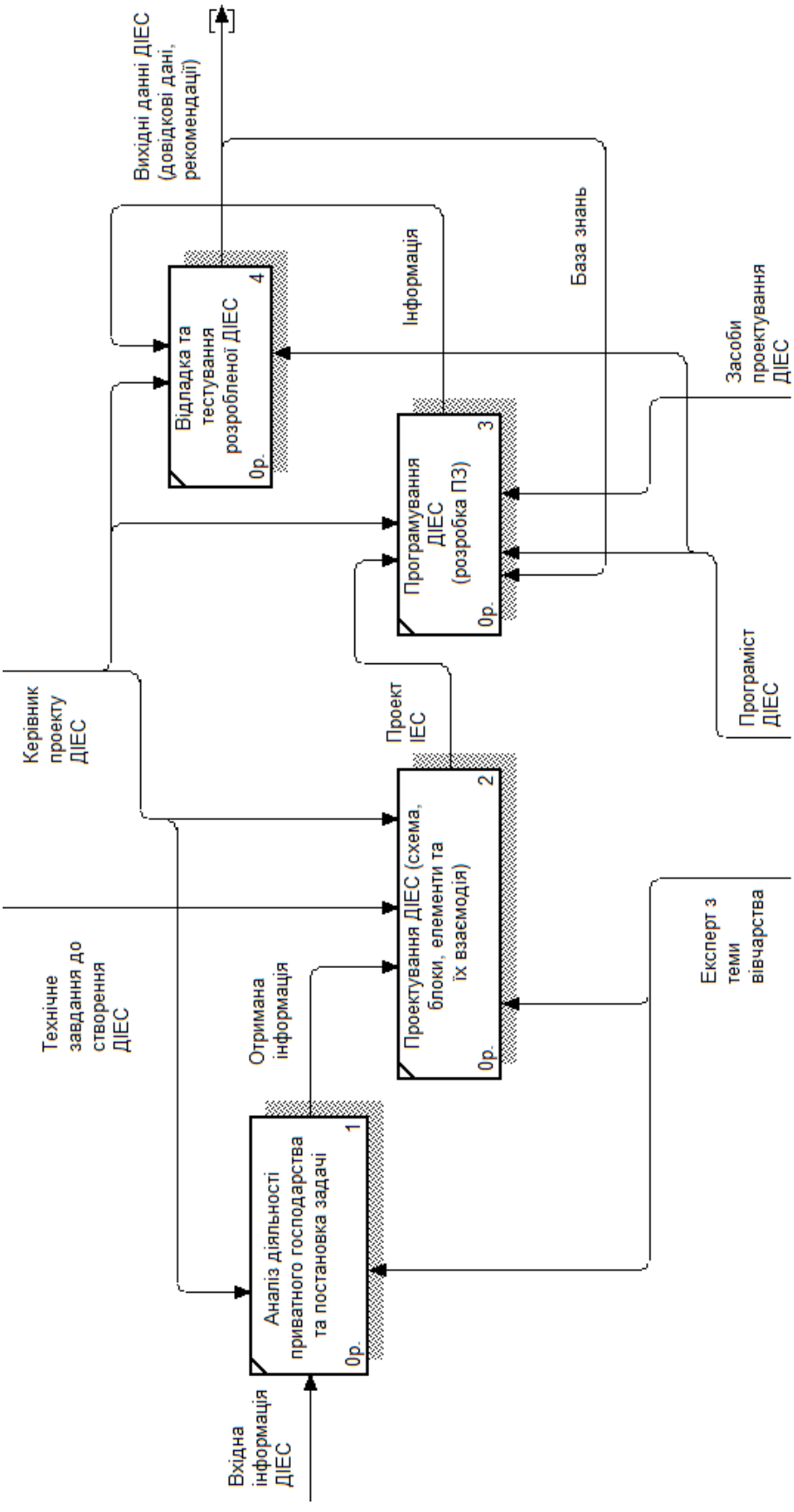

Рис. 3. Декомпозиція функціональних блоків розробленої системи

Визначимо покроково методику проектування ЕС. *Методика проектування така:*

Крок 1. Отримання технічного завдання від замовника (господарства, підприємства, фермера тощо) на розробку ДІЕС.

Крок 2. Згідно літератури та вимог до вирощування овець визначаються основні критерії для даної технології згідно технічного завдання господарства.

Крок 3. Для кожного з цих критеріїв визначаються найбільш вагомі фактори, які впливають на процес вирощування овець.

Крок 4. Визначаються основні продукційні правила, за якими буде проводитися програмування системи, а саме модулю логічної обробки знань, для даної технології. Це і є вхідними параметрами (факторами) при проектуванні ДІЕС.

Продукційні правила розробленої ІЕС:

Правило №1: якщо встановити (перемикачем на формі) технологію – «Содержание овец» у господарстві, то вказати яку саме – «пастбищно-стойловое» або «стойлово-пастбищное».

Правило №2: якщо встановити (перемикачем на формі) технологію – «Кондиция овец» у господарстві, то вказати яку саме – «высшая кондиция», «средняя кондиция» або «нижесредняя кондиция».

Правило №3: якщо встановити (перемикачем на формі) технологію – «Направление выращивания» у господарстві, то вказати яку саме – «мясное», «молочное», «мясо-молочное» або «получение шерсти».

Правило №4: якщо встановити (перемикачем на формі) технологію – «Кормление овец» у господарстві, то вказати яку саме – «ярок мясо-шерстных овец», «баранчиков мясо-шерстных овец» або «подсосных маток в лактацию».

Правило №5: якщо встановити (перемикачем на формі) технологію – «Методы разведения» у господарстві, то вказати яку саме – «чистопородное» або «скрещивание».

Крок 5. Визначаються основні вихідні критерії (фактори), які будуть виводитись після обробки вхідних правил системи на підставі правил логічного виведення, згідно відповідної технології вирощування овець.

Крок 6. Додатково (за необхідністю або за вимогою заказника) на формі системи проектуються кнопки для більш зручного її використання. Наприклад: кнопки очищення вікон від попередньої інформації, кнопка зберігання рекомендацій в окремий текстовий файл, кнопка виходу з системи тощо.

Крок 7. Тестування зробленої системи користувачами та замовником.

Крок 8. Прийняття розробленої ДІЕС замовником та її виправлення (при потребі).

Крок 9. Супроводження розробленої ДІЕС.

Проектування інтерфейсу системи.

При розробці системи було вирішено у якості автоматизованого засобу проектування застосувати Visual Studio, мова ж програмування була обрана C Sharp (C#) (Стиллмен, &Грин, 2012; Мартынов, 2007; Секунов, 2001).

Проведемо розробку інтерфейсу системи згідно отриманого технічного завдання господарства. Інтерфейс користувача (головна форма) спроектуємо у вигляді двох блоків, а саме: зліва блок вхідних даних, а справа блок вихідних даних. Знизу справа буде блок керуючих кнопок. На рис. 4 схематично зображено створений інтерфейс користувача.

<table>
<tr><td rowspan="2">Вхідні дані з технології вирощування овець</td><td>Рекомендації з технології вирощування овець</td></tr>
<tr><td>Керуючі кнопки</td></tr>
</table>

Рис. 4. Схематичне зображення інтерфейсу користувача

У блоці керуючих кнопок створимо окремі три кнопки – «Получение рекомендаций», «Сохранить в файл» та «Выход» (рис. 5). Перша кнопка буде розраховувати та виводити у окремі вікна відповідні рекомендації з вирощування овець у господарстві. Друга кнопка дозволяє зберегти отримані під час роботи експертною системи дані у текстовий файл у форматі rtf. Третя – передбачає вихід з програми.

Вимоги до відображення ДІЕС на екрані ПК: мінімальна роздільна здатність, за якої програма виглядатиме цілісно без видимих проблем – 1024×768 px.

Розглянемо структуру розробленої системи.

Блок вхідних даних (зліва на формі) має наступні елементи (рис. 5):

– «Содержание овец»: «пастбищно-стойловое» та «стойлово-пастбищное»;
– «Кондиция овец»: «высшая кондиция», «средняя кондиция» та «нижесредняя кондиция»;
– «Направление выращивания»: «мясное», «молочное», «мясо-молочное», «получение шерсти»;
– «Кормление овец»: «ярок мясо-шерстных овец», «баранчиков мясо-шерстных овец», «подсосных маток в лактацию»;
– «Методы разведения»: «чистопородное», «скрещивание».

Рис. 5. Програмний інтерфейс ДІЕС

За допомогою перемикачів відбувається вибір того чи іншого елементу відповідного меню. Також на формі існують три керуючі кнопки (рис. 5).

Кнопка «Получение рекомендаций» - розраховує та виводить у окремі вікна відповідні рекомендації з вирощування овець у господарстві (рис. 6). Кнопка «Сохранить в файл» - зберігає отримані під час роботи експертною системи дані у текстовий файл у форматі rtf (рис. 7). Третя кнопка «Выход» для виходу з програми знаходиться у нижньому правому куті головної форми експертної системи (рис. 6).

Головна форма розробленої системи наведена на рис. 6.

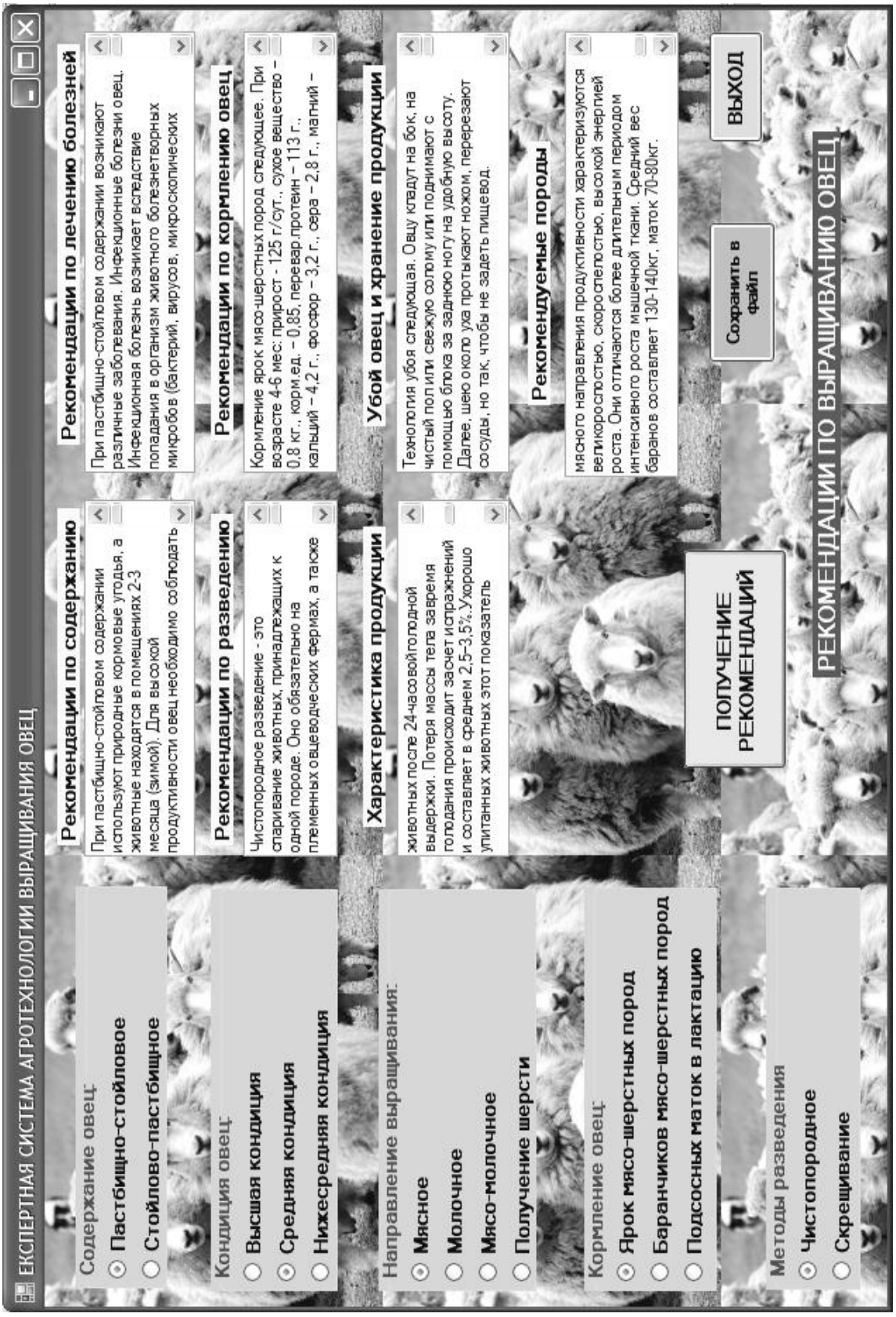

Рис. 6. Головна форма розробленої системи

Блок вихідних факторів (справа на формі) має відповідні вікна, куди виводяться відповідні до технології рекомендації, а саме (рис. 5 та рис. 6):

1. «Рекомендации по содержанию овец»;
2. «Рекомендации по лечению болезней овец»;
3. «Рекомендации по кормлению овец»;
4. «Убой овец и хранение продукции»;
5. «Рекомендуемые породы овец»;
6. «Рекомендации по разведению овец»;
7. «Характеристика продукции».

Після отримання рекомендацій бажано зберегти отримані рекомендації за допомогою відповідної кнопки на формі. При успішному збереженні файлу з'являється відповідне вікно з фразою – «Файл сохранен на диске».

Після встановлення всіх потрібних параметрів при збереженні файлу з отриманими рекомендаціями у папці Dedug розробленої системи з'явиться згенерований програмою відповідний текстовий файл у форматі rtf з ім'ям «Результаты рекомендаций» (рис. 7). Цей файл можна безпосередньо роздрукувати на принтері.

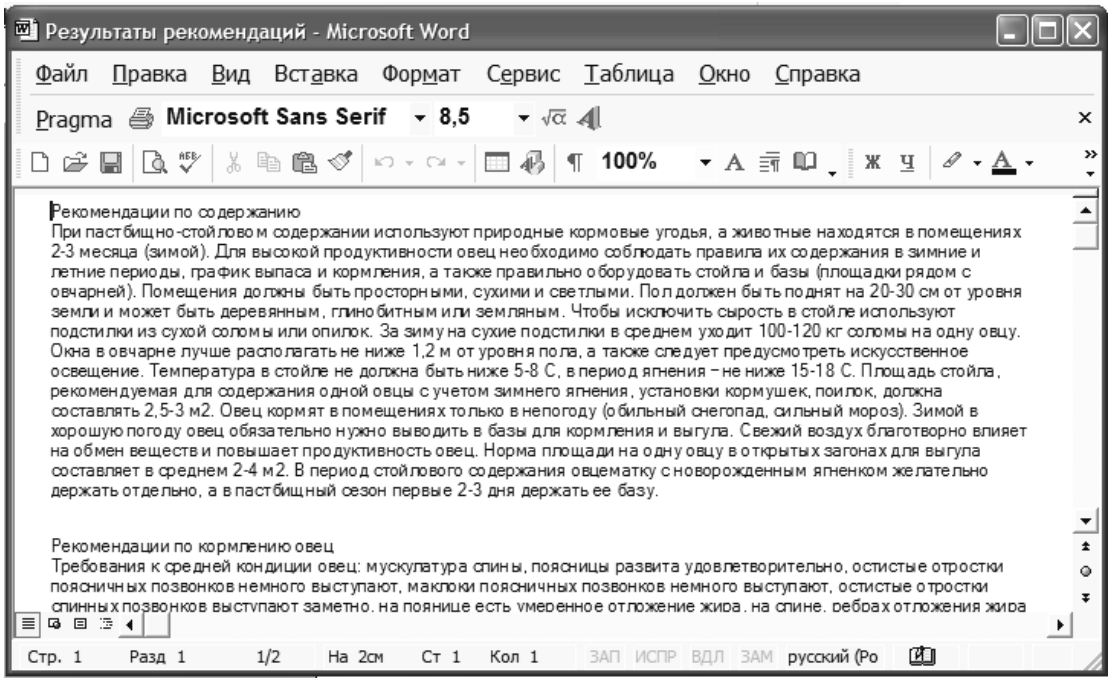

Рис. 7. Текстовий файл з рекомендаціями ДІЕС (приклад)

Вихідними правилами (рекомендаціями), що формує ДІЕС, з вирощування овець у приватних господарствах є наступні: рекомендації по утриманню овець; рекомендації по розведенню овець; характеристика продукції (отриманої з овець); рекомендації по

лікуванню хвороб овець; рекомендації по годуванню овець; забій овець та зберігання продукції; породи овець, які рекомендує система для розведення. Саме ці рекомендації і потрібні фермерам з розведення овець і саме вони визначають основну мету розробки даної ДІЕС – допомогти їм та пришвидшити пошук потрібної інформації з розведення овець надавши раціональні та ефективні дані (та методики) з технології вівчарства.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Розроблена інтелектуальна експертна система дозволяє робити запити у базу даних та виводити рекомендації та поради на екран ПК з теми технології розведення овець, а також друкувати отримані дані.

Система має дворівневу структуру, а саме – діалоговий вибір вхідних критеріїв (даних), а також модуль логічної обробки та виведення раціональних рекомендацій по технології (вихідні дані).

ДІЕС дозволяє давати рекомендації найбільш раціональної та ефективної технології вирощування овець для приватних господарств країни в залежності від вхідних критеріїв (даних).

Впровадження та використання такої системи дозволить різним приватним господарствам або окремим фермерам (вівчарям) передбачити, а у подальшому і збільшити виробництво та якість м'яса, вовни, смушок та хутра розведення овець. А це в свою чергу дозволить зменшити витрати на технологію при їх розведенні, зберегти час фермеру, при пошуку інформації для їх вирощування, заощадити на ліках, кормах, що в свою чергу підвищить прибутки та рентабельність окремого господарства.

У якості перспективи щодо розвитку розробленої системи, можна вказати те, що вона є досить широко профільною, що дозволить її використання (саму оболонку системи) для всіх господарств різного напрямку (не тільки вівчарства) та під будь-яку галузь тваринництва після її модифікації та доробки правил логічного виведення та інтерфейсу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

Джарантано, Дж., & Гари, Р. (2007). *Экспертные системы: принципы разработки и программирование*. Москва: Вильямс.

Джарантано, Дж. (2013). *Экспертные системы: принципы разработки и программирование*. Москва: Вильямс.

Козлов, А. Н. (2013). *Интеллектуальные информационные системы: учебник*. Пермь: Пермская ГСХА.

Мартынов, Н. Н. (2007). *C# для начинающих*. Киев: Кудиц-Пресс.

Мирось, В. В., & Фоминова, А. С. (2015). *Овцеводство и козоводство*. Киев: Феникс.

Нейлор, К. (2006). *Как построить свою экспертную систему*. Москва: Энерго-атомиздат.

Петров, В. Н. (2003). *Информационные системы*. Санкт-Петербург: Питер.

Ревко, П. С. (2006). Искусственные интеллектуальные системы в повседневной жизни человека. *Известия Южного федерального университета. Технические науки*, (9-2), 109-110.

Сахнюк, П. А. (2012). *Интеллектуальные системы и технологии: учеб. пособ*. Ставрополь: Агрус.

Секунов, Н. В. (2001). *Самоучитель C#*. Санкт-Петербург: БХВ-Петербург.

Стиллмен, Э., & Грин, Дж. (2012). *Изучаем C#*. Санкт-Петербург: Питер.

Таусенд, К., & Фохт, Д. (2001). *Проектирование и программная реализация экспертных систем на персональных ЭВМ*. Москва: Финансы и статистика.

Уинстон, П. (2009). *Искусственный интеллект*. Москва: Высшая школа.

Уотермен, Д. (2000). *Руководство по экспертным системам*. Москва: Мир.

Фаулер, М. (2006). UML. *Основы. Третье издание. Краткое руководство по стандартному языку объектного моделирования*. Москва: Символ-Плюс.

Черемных, С. В., Семенов, И. О., & Ручкин, В. С. (2006). *Моделирование и анализ систем. IDEF-технологии: практикум*. Москва: Финансы и статистика.

Шаров, С. В., & Хрустальов, С. О. (2016). Інструментальні засоби та етапи розробки експертних систем. *Збірник наукових праць Харківського національного університету Повітряних Сил*, 4(49), 105-109.

REFERENCES (TRANSLATED AND TRANSLITERATED)

Giarratano, J. C. & Gary, D. R. (2007). *Expert systems: Principles and Programming*. Moscow: Vilyams. (in English)

Giarratano, J. C. *Expert systems: principles of development and programming*. Moscow: Vilyams. (in Russian)

Kozlov, A. N. (2013). *Intellectual information systems: a textbook*. Perm: Permskaya GSHA. (in Russian)

Martynov, N. N. (2007). *C # for Beginners*. Kyiv: Kudits-Press. (in Russian)

Miros, V. V., & Fominova, A. S. (2015). *Sheep breeding and goat breeding*. Kyiv: Feniks. (in Russian)

Naylor, K. (2006). *How to build your own expert system*. Moscow: Energo-atomizdat. (in Russian)

Petrov, V. N. (2003). *Information systems*. St. Petersburg: Piter. (in Russian)

Revko, P. S. (2006). Artificial Intelligent Systems in everyday life of man. *News of the Southern Federal University. Technical science,* (9-2), 109-110. (in Russian)

Sachnyuk, P. A. (2012). *Intelligent Systems and Technologies: Study*. Stavropol: Agrus. (in Russian)

Secunov, N.V. (2001). *Self-C #*. St. Petersburg: BHV-Peterburg. (in Russian)

Stillman, E., & Green, J. (2012). *Learning C #*. St. Petersburg: Piter. (in Russian)

Thousend, K., & Focht, D. (2001). *Designing and software implementation of expert systems on personal computers*. Moscow: Finansy i statistika. (in Russian)

Winston, P. (2009). *Artificial Intelligence*. Moscow: Vysshaya shkola. (in Russian)

Waterman, D. (2000). *Guidance on expert systems*. Moscow: Mir. (in Russian)

Fowler, M. (2006). *UML. Fundamentals. Third edition. A Brief Guide to the Standard Object Modeling Language*. Moscow: Simvol-Plyus. (in Russian)

Cheremnykh, S. V., Semenov, I. O., & Ruchkin, V. S. (2006). *Modeling and analysis of systems. IDEF-technologies: workshop*. Moscow: Finansy i statistika. (in Russian)

Sharov, S. V., & Khrustalev, S. O. (2016). Instrumental means and stages of development of expert systems. *Collection of scientific works of Kharkiv National University of Air Forces, 4*(49), 105-109. (in Ukrainian)

Матеріал надійшов до редакції 12 липня 2017