

Національний технічний університет України
«Київський політехнічний інститут»
Кафедра обчислювальної техніки

Проект
«Експертна система»
з дисципліни «Організація баз даних»

Виконали:
студенти 2 курсу
ФІОТ гр. ІО-63
Бригада

Бабко Д.С.
Булах А.С.
Ванярха С.Е.
Гайдай А.Р.
Іщенко Б.А.
Мельник І.М.

Київ
2017 р

Зміст

1.1. Вступ.....	3
1.1.1. Мета	3
1.1.2. Основні положення.....	3
1.2. Короткий огляд продукту	11
1.2.1 Склад системи.....	11
1.2.2 Приклади анкет	11
1.3. Ділові правила і приписи	14
1.3.1. Призначення експертної системи «».....	14
1.3.2. Діаграма юзкейсів.....	14
1.3.3. Сценарій реєстрування нового користувача	15
1.3.4. Сценарій створення опитування	16
1.3.5. Сценарій додавання питання	17
1.3.6. Сценарій проходження опитування.....	18
1.3.7. Сценарій коментування опитування.....	19
1.3.8. Діаграми послідовностей.....	20
1.4. Функціональність	23
1.4.1. Можливості «Менеджер».....	23
1.4.2. Можливості «Користувач»	23
1.4.3. Можливості «Експерт»	23
1.4.4. Можливості «Аналітик»	23
1.5. Надійність	23
1.5.1. Резервне копіювання і відновлення даних	23
1.6. Джерела.....	23

1.1. Вступ

Експертна система – як це зазначено в [1] це програма, що поводить себе подібно експерту в деякій, звичайно вузькій прикладній області (типові застосування експертних систем містять у собі такі задачі, як медична діагностика, локалізація помилок в устаткуванні й інтерпретація результатів вимірів).

Експертні системи повинні вирішувати задачі, що вимагають для свого рішення експертних знань у деякій конкретній області, уміти певним чином пояснювати свою поведінку і свої рішення користувачу, так само, як це робить експерт-людина, у тій чи іншій формі експертні системи повинні мати ці знання.

Технологію побудови експертних систем будується на взаємодії автора експертної системи (інженера знань) з експертами в певній предметній області.

Основою експертної системи є база знань, яка накопичується в процесі побудови експертної системи. Знання в такій базі організовані й відображені так, щоб спростити прийняття рішення.

Однією з характеристик експертної системи є те, що вона застосовує досвід мислення найбільш кваліфікованих експертів у даній ділянці знань, що приводить до точних, творчих та ефективних рішень.

Останньою характеристикою експертної системи є те, що її можна використовувати для навчання й тренування робітників та спеціалістів.

1.1.1. Мета

Метою документа є визначення основних вимог до функціональності та експлуатаційної придатності, пред'явлених до предмета розробки.

За метою створення експертні системи можна класифікувати таким чином:

- для навчання фахівців
- для вирішення задач
- для автоматизації рутинних робіт
- для тиражування знань експертів

За завданням, що вирішується:

- Інтерпретація даних
- Діагностика
- Моніторинг
- Проектування
- Прогнозування
- Звідне Планування
- Навчання
- Керування
- Підтримка ухвалення рішень.

1.1.2. Основні положення

«Експерт» та «оцінювання»

Експерт - висококваліфікований спеціаліст, який має вищу освіту, відповідну кваліфікацію і професійні знання з питань, що досліджуються, виконує службові обов'язки, пов'язані з провадженням діяльності у відповідній галузі, безпосередньо проводить експертизу та несе персональну відповідальність за

достовірність і повноту аналізу, обґрунтованість висновків відповідно до завдання на проведення експертизи.

Основні показники експертного потенціалу:

- Високий рівень інтелекту
- Значний досвід роботи
- Визнання колег
- Активна наукова діяльність
- Існування публікацій у престижних виданнях
- Престижна освіта
- Високий особистий статус

Експертне оцінювання - особливий різновид методу опитування, де вимоги, які застосовуються в масових опитуваннях, неприйнятні. В процесі масового опитування джерелом соціологічної інформації, яка дає оцінку тим чи іншим сторонам об'єкта дослідження, виступають і представники цього ж об'єкта. При експертному опитуванні ситуація інша. Експерти майже ніколи не виступають представниками об'єкта дослідження.

На початку, заходи та процедури експертних оцінок розроблялися головним чином для застосування футурологічної діяльності (передбачення, прогнозування), економічного планування, питань управління та наукознавства. Методологічна обумовленість застосування експертних оцінок в соціології найбільш проявляється при розгляді діагностичного аспекту експертизи. Тому, як правило, експертне оцінювання (опитування) широко використовується в розвідувальних та пробних дослідженнях для отримання попередніх даних про об'єкт, предмет аналізу, для уточнення гіпотез та завдань основного дослідження, для розробки прогнозу і доповнення та з'ясування певних процесів та явищ. Як правило, досліджують і такі ситуації, що пов'язані зі спробою прогнозувати зміну того чи іншого соціального явища, процесу, з необхідністю уявити стан предмета, який становить інтерес, або дати оцінку таким сторонам діяльності і характеристикам людей, за якими їхня самооцінка може виявитися хибною. Подібна інформація може надходити тільки від компетентних осіб - експертів, які мають глибокі знання про предмет або об'єкт дослідження.

Практика застосування експертних методик (як для вирішення проблем прогнозування, управління, так і в традиційних сферах соціологічного дослідження) показала, що вони такою ж мірою і важливі і надійні, як і метод масових опитувань з його розширеною методикою забезпечення репрезентативності. Експертні методики містять ряд експліційованих (спільних, схожих) понять: експертні системи, діагностика, експертне опитування та оцінювання, думки, сценарії. Але в соціології традиційно в експертизі акцентується на ролі опитування. Таким чином, експертне опитування посідає тут центральне місце.

Важливий момент при проведенні такого роду опитування - відбір експертів, передусім за рівнем їх компетентності. На першому етапі відбору як критерій доцільно використовувати дві ознаки: рід занять та стаж діяльності за профілем, що нас цікавить. За необхідності потрібно брати до уваги також і рівень, характер освіти, досвід, вік. Як завжди, перший список може бути досить широким, однак у подальшому його доцільно "звужити", оскільки не кожна людина здатна виступити в ролі експерта. Але основним серед усіх критеріїв відбору експертів

є їх компетентність. Для її визначення застосовуються - з тим чи іншим ступенем точності - методи самооцінки експертів, оцінки результатів минулої діяльності кандидатів у експерти та колективні оцінки авторитетності експертів.

Метод самооцінки обумовлює оцінку кожним із кандидатів в експерти власної компетентності (як правило, за 5-бальною шкалою), включаючи показники, які характеризують ступінь його участі в досліджуваній проблемі. Використовують також різного роду довідники, типу "хто є хто", списки членів професійних груп і т. ін. Це допомагає визначити, де даний кандидат навчався, працював, якими аспектами професійної діяльності він займався. Критеріями відбору спеціалістів до експертної групи може служити також і кількість опублікованих праць і т. ін. Якщо кандидати в експерти знають один одного, то ефективним може бути метод колективної оцінки кожного із них іншими. Хоча деякі спеціалісти вважають, що колективні оцінки менш точні, ніж самооцінки, все-таки аналіз розходжень між ними дозволяє здійснити більш обумовлений підбір експертів.

Не менш ефективним виявляється метод "снігового нальоту": відібрані на першому етапі експерти називають компетентних спеціалістів, котрі, в свою чергу, включаються в процес відбору; ці ж відібрані експерти рекомендують інших спеціалістів. Цей процес завершується стабілізацією сукупності рекомендованих експертів, коли їх список перестає поповнюватися новими прізвищами. Після складання попереднього списку проводиться їх оцінка шляхом соціометричної процедури, де критерієм оцінки спеціаліста є частота рекомендацій його іншими експертами.

До експертної групи повинні входити спеціалісти, які здатні провести експертизу із врахуванням різних сторін досліджуваної проблеми. Тому до експертної групи бажано залучати спеціалістів, котрі представляють різні наукові напрями.

У прикладній соціології розроблено цілий ряд прийомів опитування експертів, які використовуються для отримання різного роду оцінок, в тому числі і прогностичних. Основний інструментарій експертних опитувань – анкета або бланк-інтерв'ю, які розроблені за спеціальною програмою. Відповідно до цього процедура опитування полягає або в анкетуванні, або в інтерв'юванні експертів. На відміну від масового опитування, програма опитування експертів не настільки деталізована і має переважно концептуальний характер. В ній, передусім, однозначно формулюють явища, які підлягають вивченню, та передбачаються у вигляді гіпотез можливі варіанти їх наслідків.

Продуктивним способом організації групової експертної оцінки є колективне обговорення індивідуальної думки або виділення із експертної групи найбільш кваліфікованих авторитетних експертів з метою корекції оцінок. Суттєвим моментом узгодження думок в процесі групового вибору є селекція найбільш мотивованих висновків. Експертне оцінювання – це не просто формулювання, відбір і обробка висновків експертів. Воно визначається насамперед роботою експертів щодо забезпечення аргументації висновків, їх мотивації та демонстрації того, чому надається перевага. Висновки (думка) - це свого роду ідейна позиція в системі різних варіантів особистої інформації, аналіз якої сприяє висуненню гіпотез та встановленню істини.

Експертне опитування передбачає дотримування таких правил та вимог:

- вказати, сутність якої проблеми вивчається, і мотив звернення до експерта;
- дані, що підтверджують компетентність експерта (галузь знань, стан, кваліфікація);
- інформація про зміст пропонованих питань;
- правильне формулювання кожного проблематичного питання;
- допоміжні зауваження, коментарі, пропозиції.

Від якості конкретних експертних оцінок залежить надійність рішень, що приймаються на їх основі, що великою мірою залежить і від якісного складу експертів, ретельності розробки процедури експертизи, рівня проведення опитування та обробки її результатів.

Методи експертного оцінювання.

Індивідуальні експертні методи – це використання думок експертів, які сформульовані особисто кожним із них самостійно без врахування думок інших експертів. До індивідуальних експертних методів належать: інтерв'ю та анкетування[2, 3].

- Метод інтерв'ю полягає у постановці аналітиком під час бесіди з експертом запитань про чинники, що визначають стан досліджуваного об'єкта, шляхи і засоби зміни стану об'єкта у бажаному напрямі. Зміст запитань визначає заздалегідь складена програма, яку можна уточнювати у процесі інтерв'ю. Ефективність цього методу визначається: глибиною аналізу розглянутої проблеми, який передував опитуванню; якістю програми опитування; методикою її проведення; компетентністю обраних експертів для вирішення проблеми.
- Метод анкетування ґрунтується на самостійній підготовці експертом відповідей на запитання анкети. Однак досвід свідчить, що у письмовому викладі великого значення набувають такі суб'єктивні фактори, як “місцевий патріотизм”, небажання виступати з критикою товаришів по службі і керівників, скептицизм щодо значення і способів дослідження, неправильне тлумачення чи нерозуміння запитання, звичайне небажання займатися невласливою опитуваному роботою. Усе це негативно позначається на якості аналізу, проведеного за допомогою анкетування.

Основні переваги індивідуальних методів полягають у можливості використання здібностей і знань окремого експерта, а також у відносній простоті проведення цільового аналізу. Основний їхній недолік - обмеженість знань кожного з опитуваних про стан і розвиток суміжних сфер діяльності.

Колективні експертні методи – це методи, які забезпечують формування єдиної спільної думки в результаті взаємодії залучених фахівців-експертів. До індивідуальних експертних методів належать: метод комісії, методи Дельфі, відстороненого оцінювання, суду.

Експертне оцінювання на основі *методу комісії* (після відкритої дискусії шляхом голосування виробляється єдина думка експертів) дає змогу виробити найкращу альтернативу для оцінювання конкретної ситуації з урахуванням дії

комплексу якісно різних факторів. Недолік методу, окрім відсутності анонімності, полягає ще й у тому, що група експертів, які беруть участь у нарадах і висловлюють свої судження, керується здебільшого логікою компромісу. При цьому в учасників остаточно сформувалися оцінки і варіанти вирішення проблеми, які не обов'язково ліпші за ті, що висловлені на нараді.

Метод “відстороненого оцінювання” спрямований на усунення вказаного недоліку методу комісій. Відповідно до цієї методики робота поділена на два періоди: вільного висловлення ідей і критичного аналізу. При цьому обговорення організують так, щоб висловлене експертом первинне судження не обтяжувало його, а стимулювало до подальшої роботи з аналізу і підготовки рішень.

Іншим прикладом колективного методу організації експертиз є **метод суду**. Експертна комісія розбивається на три частини. Перша - виступає в ролі захисників досліджуваної ініціативи, друга - намагається виявити негативні сторони, а третя, за аналогією з судовим засіданням, - регулює хід експертизи та виносить остаточне рішення.

Відмовитись від незручних для експертів форм роботи (наприклад, дискусії за “круглим столом” чи інших видів обговорення протилежних точок зору) дає можливість застосування **методу Делфі**. Він характеризується такими основними властивостями: анонімністю, регульованим зворотним зв'язком і груповою відповіддю. Анонімність забезпечується спеціальною формою опитувального листка чи особливими прийомами опитування, наприклад, з використанням сучасних мультимедійних технологій чи комп'ютерної техніки. Опитування зазвичай проводять у 3-4 тури, на кожному з яких за допомогою загальновідомих статистичних методів визначають групову оцінку. Регульованого зворотного зв'язку досягають завдяки тому, що учасникам експертизи після кожного туру пропонують ознайомитися із колективним судженням, пояснити свою можливу незгоду з ним і навіть дають можливість переглянути власну первинну оцінку.

Анкета

В деяких методах експертного оцінювання використовують анкети. **Анкета [4]** - структурно організований набір запитань, кожне з яких логічно пов'язане з основною метою дослідження.

Якість, отриманої інформації залежить, значною мірою, від того, яким чином скомпонована або структурована анкета. Побудова анкети повинна складатися з чотирьох частин:

- **Вступна:**

У першій частині анкети пояснюється мета дослідження, гарантується анонімність відповідей, дається інструкція щодо заповнення, підкреслюються важливість та цінність відповідей респондента.

- **Демографічна:**

В цій частині фіксуються вікові, статеві, етнічні, соціально-статусні, професійні, освітні, сімейно-шлюбні, соціально-поселенські та інші ознаки респондента.

- **Основна:**

Складається з набору запитань, відповіді на які передбачають виконання завдань дослідження. Ця частина анкети за змістом найбільш вагома та складна.

Запитання повинні враховувати психологічні особливості респондентів. Запитання складають блоками, у чіткій логічній послідовності. Головна мета запитань анкети полягає в тому, щоб зацікавити респондента, включити його поступово в процес співпраці з анкетною. Далі пропонуються запитання складніші, які виявляють соціальні установки та орієнтації респондента, оцінки та судження, які мають прямий стосунок до основної теми дослідження. У заключній частині тексту анкети пропонуються найбільш інтимні та різного характеру контрольні запитання, мета яких — поглибити та уточнити інформацію, яка була отримана у відповідях на попередні запитання.

- *Завершальна:*

Остання частина анкети передбачає подяку респондентам та, можливим є, побажання на майбутнє.

Види запитань в анкетах.

1. За формою відповідей:

- закриті (містять повний набір можливих варіантів відповіді):
- дихотомічні (відповіді так/ні)
- полі варіативні (є декілька варіантів відповіді)
- шкальні (наприклад, повністю згоден/згоден/частково згоден/не згоден)
- табличні
- напівзакриті (містять певні варіанти відповіді, але можна також вказати власну);
- відкриті (передбачають самотійну та розгорнуту відповідь респондентів):
 - особистісні (стать, вік, професія, освіта і тд);
 - про факти свідомості (думки, враження, судження, мотиви і тд);
 - про факти поведінки (результати діяльності, вчинки, дії і тд).

2. За сформованістю запитання:

- прямі (спрямовані на безпосереднє та відкрите отримання інформації від респондентів);
- непрямі (пов'язані з використанням якоїсь уявної ситуації, яка дозволить приховано дізнатись певну інформацію)

3. За функціями:

- інформаційні (основні);
- фільтри (використовують у випадку, якщо потрібно зібрати інформацію не від усіх респондентів, а від певної категорії);
- контрольні (дають можливість уточнити правильність відповіді респондента).

Результатом роботи експертної системи

Результатом роботи експертної системи є статистичні данні, які систематизують для [5]:

- Побудови, або поповнення моделі властивостей об'єкта на основі оцінок експертів і їх аргументації.
- Оцінити узгодженість думок експертів. За відсутності значимої узгодженості експертів необхідно виявити причини неузгодженості (наявність груп), оцінити помилку дослідження.

Також до позитивних сторін результатів роботи експертної системи можна віднести нівелювання помилок окремих експертів і вироблення єдино-правильної позиції чи прийняття кваліфікованого рішення.

Але, слід зауважити, що метод експертних оцінок не може замінити ні адміністративних, ні планових рішень, він лише дозволяє поповнити інформацію, необхідну для підготовки і прийняття таких рішень. Використання результатів експертних оцінок доцільно тільки там, де неможливо застосувати більш точні методи.

На сьогоднішній день склалася певна технологія розробки експертних систем, що включає 6 етапів:

- Етап 1. Ідентифікація

Визначаються завдання, які підлягають вирішенню. Планується хід розробки прототипу експертної системи, визначаються: потрібні ресурси (час, люди, ЕОМ і т.д.), джерела знань (книги, додаткові фахівці, методики), наявні аналогічні експертні системи, мети (поширення досвіду, автоматизація рутинних дій і ін.), класи розв'язуваних завдань і т.д. Етап ідентифікації - це знайомство і навчання колективу розробників. Середня тривалість 1-2 тижні.

На цьому ж етапі розробки експертних систем проходить витяг знань. Інженер по знаннях допомагає експерту виявити і структурувати знання, необхідні для роботи експертної системи, з використанням різних способів: аналіз текстів, діалоги, експертні ігри, лекції, дискусії, інтерв'ю, спостереження та інші. Вилучення знань - це отримання інженером по знаннях більш повного уявлення про предметну область і методи прийняття рішення в ній. Середня тривалість 1-3 місяці.

- Етап 2. Концептуалізація

Виявляється структура отриманих знань про предметну область. Визначаються: термінологія, перелік основних понять і їх атрибутів, структура вхідної та вихідної інформації, стратегія прийняття рішень і т.д. Концептуалізація - це розробка неформального опису знань про предметну область у вигляді графа, таблиці, діаграми або тексту, яке відображає основні концепції і взаємозв'язки між поняттями предметної області. Середня тривалість етапу 2-4 тижні.

- Етап 3. Формалізація

На етапі формалізації всі ключові поняття і відносини, виявлені на етапі концептуалізації, виражаються на деякій формальній мові, запропонованому (обраному) інженером по знаннях. Тут він визначає, чи підходять наявні інструментальні засоби для вирішення даної проблеми або необхідний вибір іншого інструментарію, або потрібні оригінальні розробки. Середня тривалість 1-2 місяці.

- Етап 4. Реалізація

Створюється прототип експертної системи, що включає базу знань і інші підсистеми. На даному етапі застосовуються такі інструментальні засоби: програмування на звичайних мовах (Java, C та ін.). Четвертий етап розробки експертних систем в якійсь мірі є ключовим, тому що тут відбувається створення програмного комплексу, який демонструє життєздатність підходу в цілому. Середня тривалість 1-2 місяці.

- Етап 5. Тестування

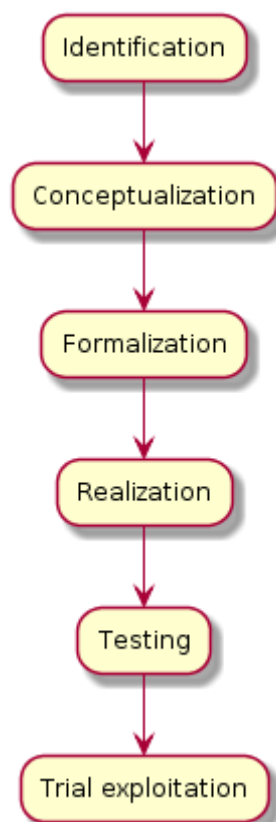
Прототип перевіряється на зручність і адекватність інтерфейсів вводу-виводу, ефективність стратегії управління, якість перевірочних прикладів, коректність бази знань. Тестування - це виявлення помилок в обраному підході, виявлення помилок в реалізації прототипу, а також вироблення рекомендацій з доведення системи до промислового варіанту.

- Етап 6. Дослідна експлуатація

Перевіряється придатність експертної системи для кінцевих користувачів. За результатами цього етапу може знадобитися суттєва модифікація експертної системи.

Процес розробки експертної системи не зводиться до суворой послідовності перерахованих вище етапів. В ході робіт доводиться неодноразово повертатися на більш ранні етапи і переглядати прийняті там рішення.

Діаграма Активності:



1.2. Короткий огляд продукту

1.2.1 Склад системи

Експертна система складається з:

- бази знань призначеної для зберігання вихідних і проміжних фактів в робочій пам'яті і зберігання моделей і правил маніпулювання моделями в базі правил;
- організатора опитувань, який забезпечує організацію опитування, його структуру та взаємозв'язки з іншими компонентами;
- компонента проведення опитування, який забезпечує реалізацію опитування на базі інших компонентів експертної системи(бази знань та організатора опитувань).

1.2.2 Приклади анкет

Популярність ігор

1. Як часто Ви граєте в ігри?

•	Кожен день
	Декілька разів на тиждень
	Декілька разів на місяць
	Декілька разів на рік

2. Якому виду ігор ви надаєте перевагу?

•	Комп'ютерні
•	Настільні
•	Карткові
	Навчальні
•	На відкритому просторі
	Інші (відкрита відповідь)

3. Ви граєте в наступні комп'ютерні ігри?

	так	Час від часу	ні
Adventury	•		
Бойовики		•	
Загадки	•		
Логічні	•		
Онлайн (MMORPG)	•		
RPG	•		
Ігри-симулятори		•	
Спортивні			•
Стратгії	•		
Гонки			•

4. Як Ви зазвичай отримуєте комп'ютерні ігри?

	Купую
•	Беру на прокат
	Качаю з інтернету

5. Ви граєте на таких пристроях?

	Так	Час від часу	Ні
Комп'ютер	•		
Ноутбук	•		
Мобільний телефон		•	
Консоль			•

6. Комп'ютер vs. консоль?

•	Комп'ютер
	Консоль
	Жоден з варіантів

7. PlayStation vs. Xbox?

	PlayStation
	Xbox
•	Жоден з варіантів

8. Назвіть свою улюблену гру:

9. Яку суму на рік Ви витрачаєте на ігри?

•	0 - 500 грн
	501 - 1000 грн
	1 001 - 2 500 грн
	Більше

10. Вкажіть вашу стать:

•	Чоловіча
	Жіноча

Оцінка кафе

1. Як часто ви відвідуєте кафе?

	Щодня
	Декілька разів на тиждень
	Раз в тиждень
•	Декілька разів на місяць
	Раз в місяць
	Декілька разів на рік
	Вперше

2. Зазначте настільки ви згодні з наступними висловлюваннями?

	Повністю погоджуюсь	Погоджуюсь	Не погоджуюсь
Закуси були поданні відмінно	•		
Закуси були якісні і смачні		•	
Ціна була приваблива		•	
Інтер'єр кафе приємний		•	
Wi-Fi підключення достатньо швидке і якісне		•	

3. Відсортуйте від найкращого до найгіршого

3	Інтерєр
4	Обслуговування
6	Смак страв
5	Смак напоїв
2	Чистота у вбиральні
1	Місце розташування закладу

4. Оцініть наше кафе

1*	2*	3*	4*	5*
			•	

5. Чи порекомендували б ви наше кафе

•	Так
	Ні

6. Вкажіть вашу стать:

•	Чоловіча
	Жіноча

7. Вкажіть вашу вікову групу

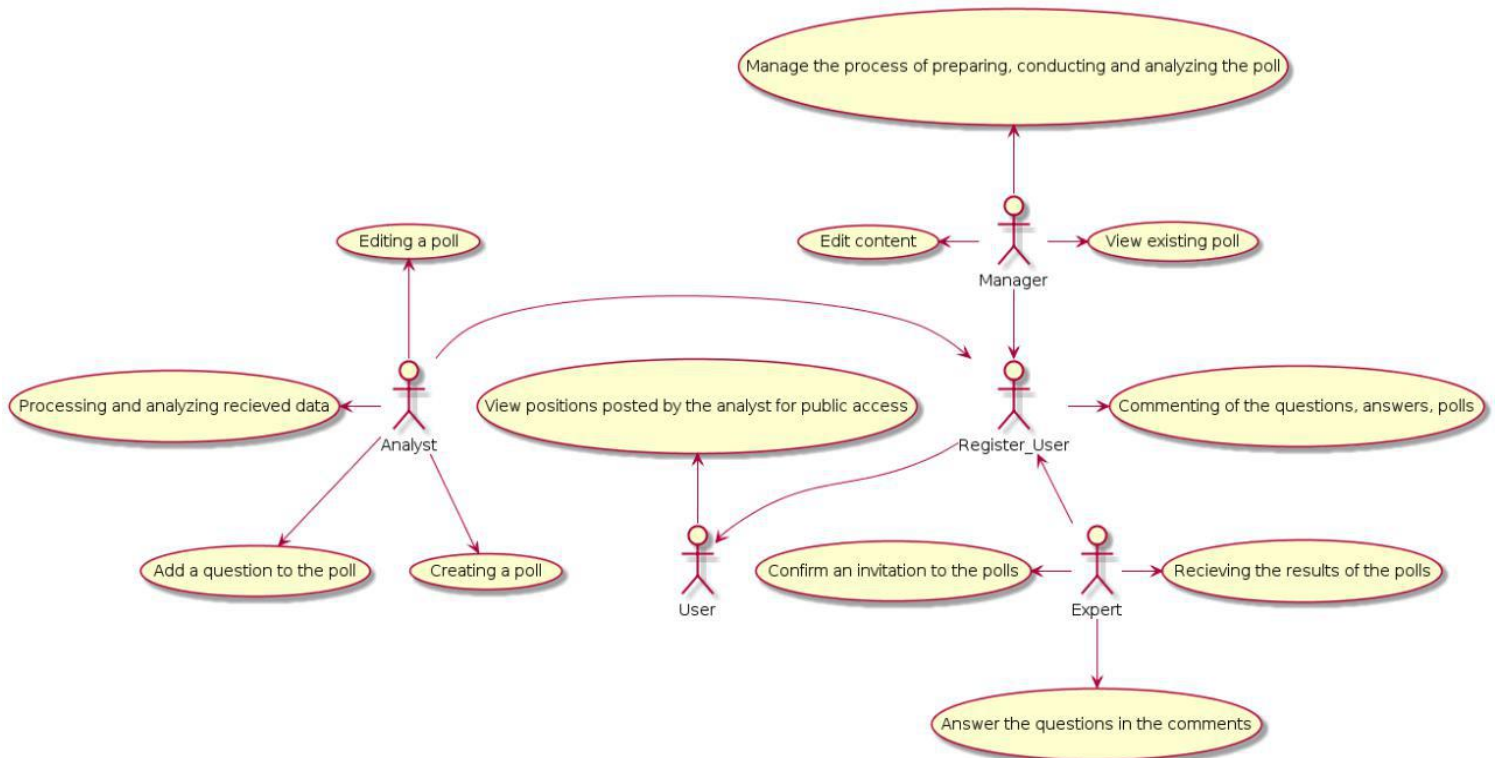
•	<20
	21-30
	31-40
	41-50
	51-60
	60+

1.3. Ділові правила і приписи

1.3.1. Призначення експертної системи «»

Функціональність системи призначена для полегшення проведення опитувань, отримання результатів у реальному часі. Легко створювані користувачем опитування, анкети, з допомогою яких можна керувати такими аспектами бізнесу, як рівень задоволення клієнтів та степінь зацікавленості клієнтів та співробітників. Категорію людей, серед яких можна проводити тести можна вибирати.

1.3.2. Діаграма юзкейсів



1.3.3. Сценарій реєстрування нового користувача

Ідентифікатор:	ES-1.1
Назва:	Реєстрація нового користувача
Учасники:	Користувач; Система
Передумови:	Експерт отримав запрошення на реєстрацію і проходження опитування
Результат:	Створення нового облікового запису користувача
Виключні ситуації:	Експерт відхилив запрошення на участь
Основний сценарій:	<ol style="list-style-type: none">1. Користувач натискає кнопку [Реєстрація].2. Система пропонує форму «Реєстрація нового користувача».3. Користувач заповнює поля: логін, пароль, прізвище, ім'я, e-mail, сфера діяльності, рівень обізнаності.4. Користувач підтверджує правильність введених даних.5. Система перевіряє введені дані.6. На електронну пошту, вказану користувачем при реєстрації, система посилає листа з посиланням для підтвердження адреси електронної пошти.7. Після проходження за посиланням користувач стає повноправним користувачем системи.

1.3.4. Сценарій створення опитування

Ідентифікатор:	ES-2.1
Назва:	Створення опитування
Учасники:	Аналітик; Система; Експерт; Менеджер
Передумови:	Поява нової заявки на опитування у менеджера
Результат:	Створення опитування та надсилання запрошень експертам
Виключні ситуації:	Аналітик перервав процес створення опитування
Основний сценарій:	<ol style="list-style-type: none">1. Аналітик натискає кнопку [Створити опитування]2. Система пропонує форму «Створення нового опитування»3. Аналітик заповнює поля: Назва опитування4. Система пропонує форму «Редагування опитування»5. Аналітик коригує опитування, додає питання, «особливі» питання та натискає кнопку [Завершити створення]6. Система завершує опрацювання опитування та надсилає запрошення на участь експертам7. Система надсилає повідомлення щодо створення опитування Менеджеру

1.3.5. Сценарій додавання питання

Ідентифікатор:	ES-2.2
Назва:	Додавання питання
Учасники:	Аналітик; Система
Передумови:	Створення нового опитування Аналітиком
Результат:	Опитування, яке містить питання
Виключні ситуації:	Аналітик перервав процес додавання питання
Основний сценарій:	<ol style="list-style-type: none">1. Аналітик натискає кнопку [Додати питання]2. Система пропонує форму «Додавання нового питання»3. Аналітик вибирає тип питання4. Система пропонує форму питання згідно обраному типу5. Аналітик заповнює поля: Питання, Відповіді на питання6. Система пропонує форму «Параметри питання»7. Аналітик коригує питання та натискає кнопку [Зберегти питання]8. Система завершує опрацювання створеного питання та повертає Аналітика до форми «Створення опитування»

1.3.6. Сценарій проходження опитування

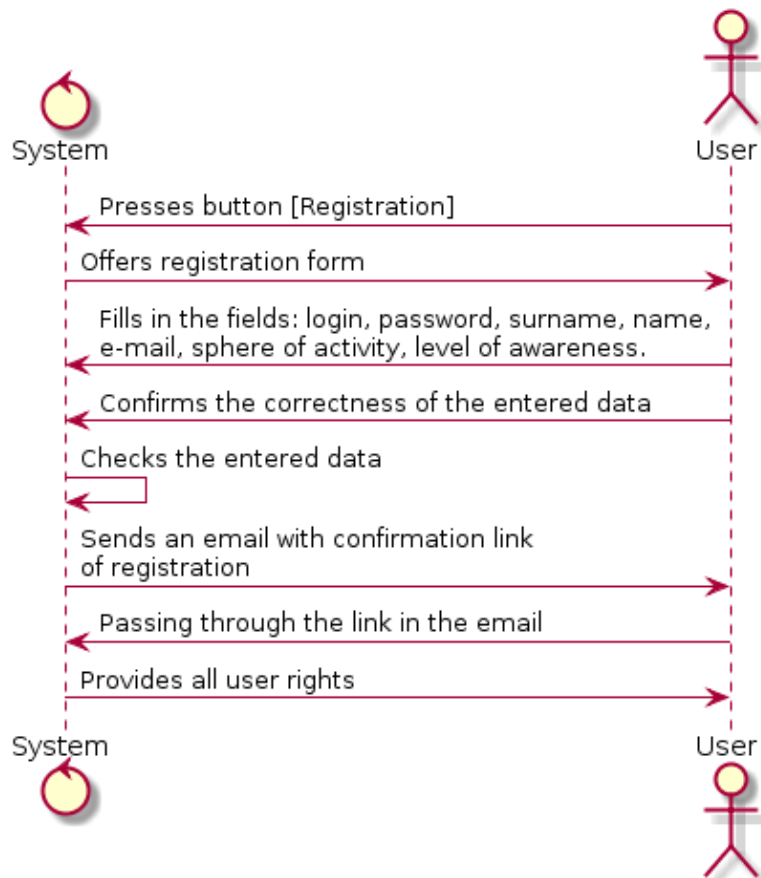
Ідентифікатор:	ES-2.3
Назва:	Створення опитування
Учасники:	Аналітик; Система; Експерт; Менеджер
Передумови:	Поява нової заявки на опитування у менеджера
Результат:	Створення опитування та надсилання запрошень експертам
Виключні ситуації:	Аналітик перервав процес створення опитування
Основний сценарій:	<ol style="list-style-type: none">1. Аналітик натискає кнопку [Створити опитування]2. Система пропонує форму «Створення нового опитування»3. Аналітик заповнює поля: Назва опитування4. Система пропонує форму «Редагування опитування»5. Аналітик коригує опитування, додає питання, «особливі» питання та натискає кнопку [Завершити створення]6. Система завершує опрацювання опитування та надсилає запрошення на участь експертам7. Система надсилає повідомлення щодо створення опитування Менеджеру

1.3.7. Сценарій коментування опитування

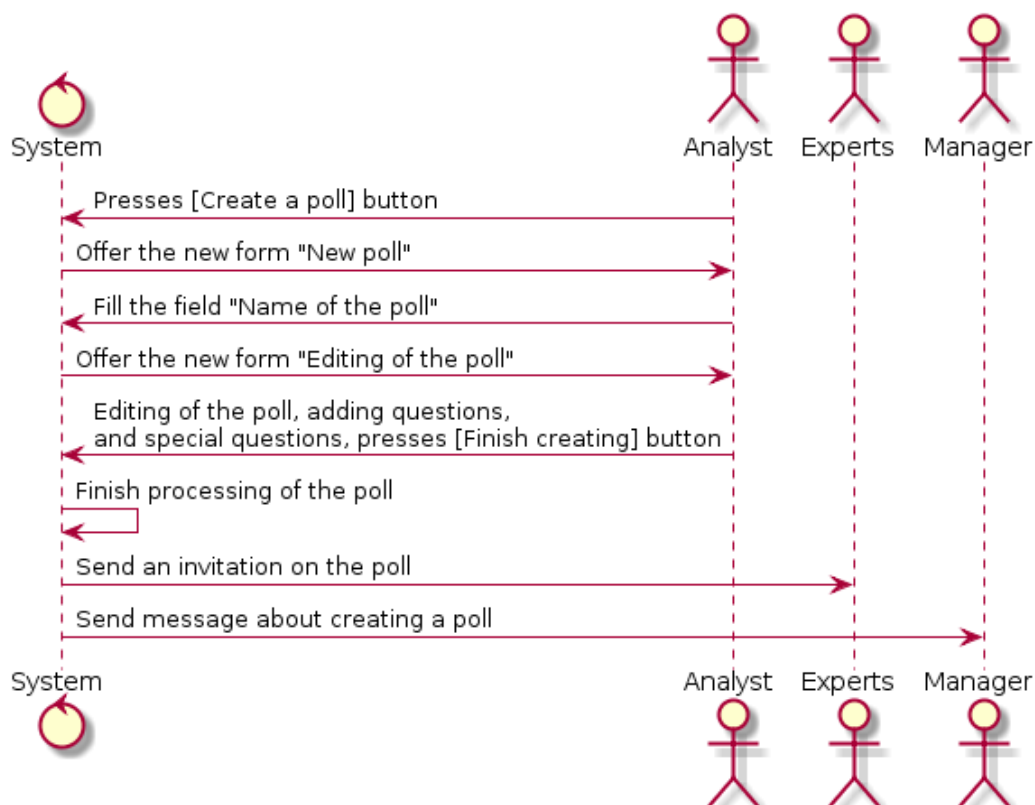
Ідентифікатор:	ES-2.4
Назва:	Проходження опитування
Учасники:	Користувач={ Експерт;Аналітик;Менеджер }; Система
Передумови:	Користувач побажав залишити коментарій до опитування
Результат:	Опитування отримає новий коментарій
Виключні ситуації:	Експерт Аналітик Менеджер перервав процес коментування опитування
Основний сценарій:	<ol style="list-style-type: none">1. Користувач натискає кнопку [Останні опитування];2. Система пропонує форму «Останні опитування»;3. Користувач обирає опитування;4. Система пропонує форму «Перегляд опитування»;5. Користувач читає інформацію та натискає кнопку [Коментарі];6. Система пропонує форму «Коментарі»7. Користувач читає вже написанні коментарі та у текстовому полі знизу може написати власний коментар та натискає кнопку надіслати коментар

1.3.8. Діаграми послідовностей

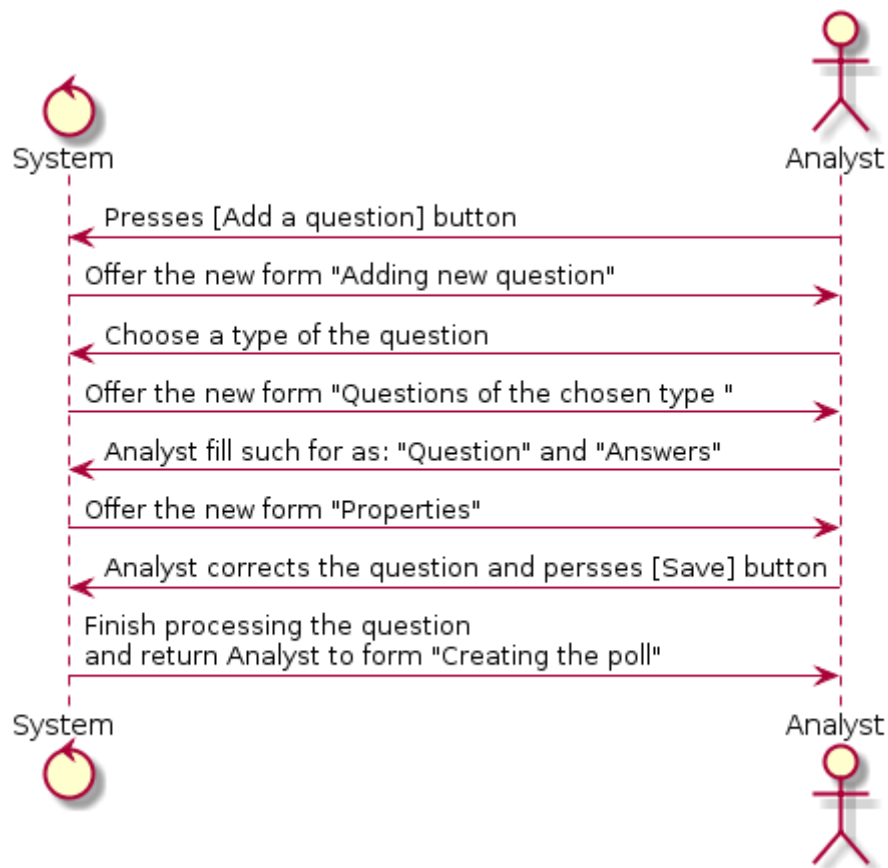
1.3.6.1 Діаграма ES – 1.1



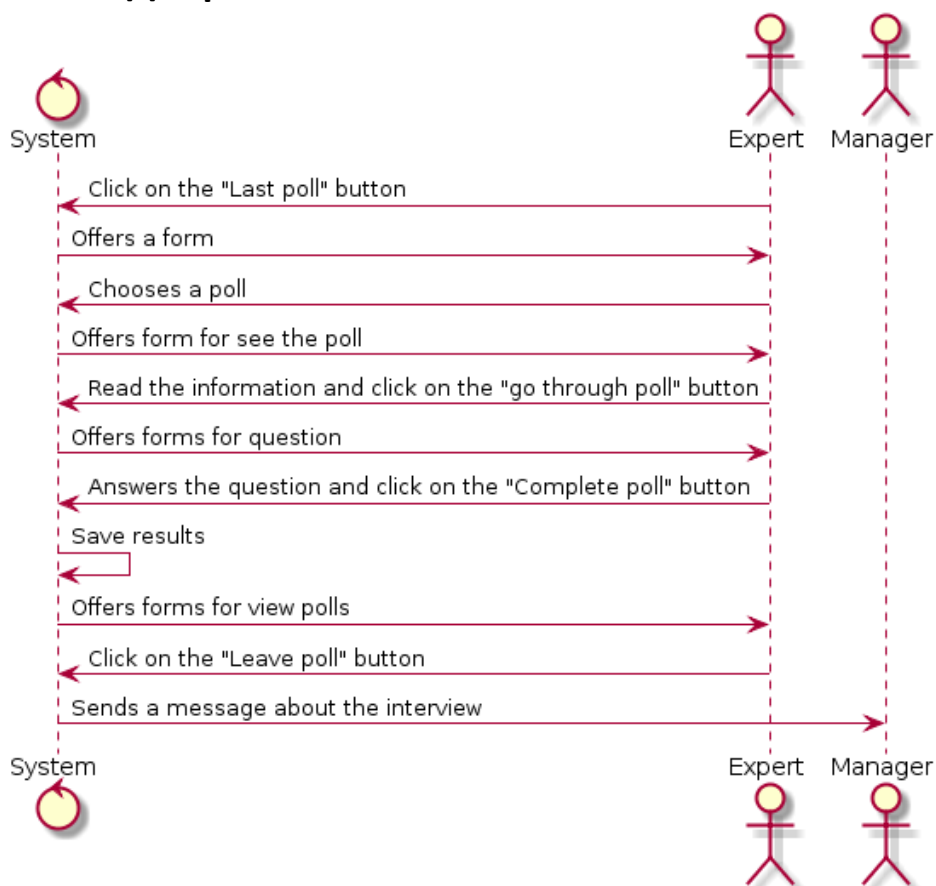
1.3.6.2 Діаграма ES – 2.1



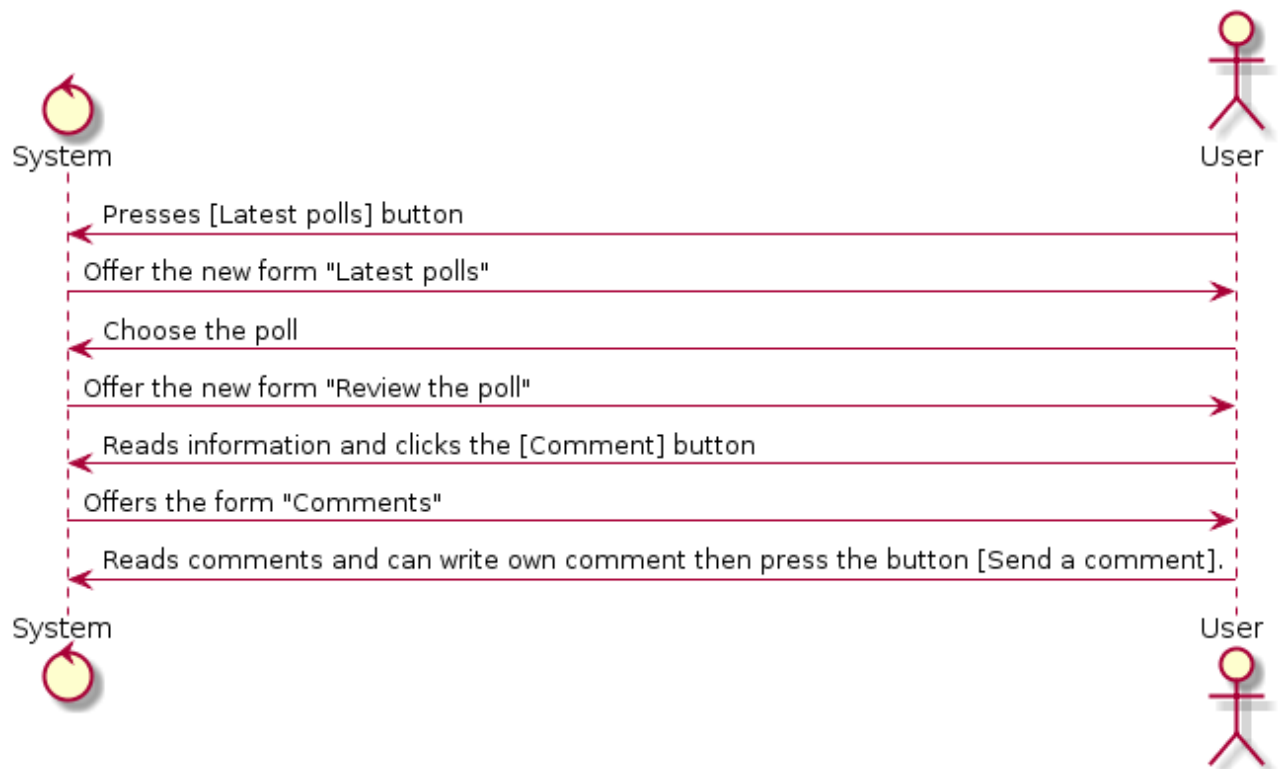
1.3.6.3 Діаграма ES – 2.2



1.3.6.3 Діаграма ES – 2.3



1.3.6.4 Діаграма ES – 2.4



1.4. Функціональність

Основні вимоги до функціональності, що пред'являються зацікавленими особами до предмета розробки, відносяться до чотирьох категорій:

- Менеджер
- Користувач
- Експерт
- Аналітик

1.4.1. Можливості «Менеджер»

- Проглядати існуючі опитування;
- Коментувати питання, відповіді, опитування;
- Отримувати результати опитувань;
- Керувати процесом підготовки, проведення та аналізування опитування
- Редагувати/коригувати контент.

1.4.2. Можливості «Користувач»

- Переглядати положення виставлені аналітиком для публічного доступу.

1.4.3. Можливості «Експерт»

- Підтверджувати запрошення на опитування;
- Відповідати на питання в опитуваннях;
- Коментувати питання, відповіді, опитування;
- Отримувати результати опитувань.

1.4.4. Можливості «Аналітик»

- Створювати опитування;
- Редагувати опитування;
- Оброблювати та аналізувати отриманні результати;
- Коментувати питання, відповіді, опитування.

1.5. Надійність

1.5.1. Резервне копіювання і відновлення даних

Регулярне здійснення резервного копіювання баз даних.

1.6. Джерела

[1] http://stn.ucoz.ru/index/ekspertni_ta_navchalni_sistemi/0-287

[2] Навчальний посібник «Теорія економічного аналізу» Автор: **Купалова Г.І.** Видавництво: «ЗНАННЯ» Рік: **2008** Сторінки: **639**

[3] Методологія експертного оцінювання: конспект лекцій / уклад. : В. П. Новосад, Р. Г. Селіверстов. - К. : НАДУ, 2008. - 48 с.

[4]<https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%BA%D0%B5%D1%82%D0%B0>

- [5] <http://ukrbukva.net/page,10,64178-Metody-ekspertnyh-ocenok.html>
- [6] <http://www.aiportal.ru/articles/expert-systems/development-expsys.html>
- [7] <https://uk.wikipedia.org/wiki/%D0%95%D0%BA%D1%81%D0%BF%D0%B5%D1%80%D1%82>
- [8] http://pidruchniki.com/12590605/sotsiologiya/ekspertne_otsinyuvannya
- [9] <https://www.surveymonkey.ru/>


```

DROP TABLE IF EXISTS alternatives;
DROP TABLE IF EXISTS variants;
DROP TABLE IF EXISTS answers;
DROP TABLE IF EXISTS metadata;
DROP TABLE IF EXISTS questions;
DROP TABLE IF EXISTS question_types;
DROP TABLE IF EXISTS sessions;
DROP TABLE IF EXISTS experts;
DROP TABLE IF EXISTS polls;

CREATE TABLE polls (
    id INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY
(id),
    title VARCHAR(64) NOT NULL,
    description VARCHAR(1024)
);

CREATE TABLE question_types (
    id INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY
(id),
    name VARCHAR(32) NOT NULL
);

CREATE TABLE questions (
    id INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY
(id),
    qindex INTEGER NOT NULL,
    qtext VARCHAR(512) NOT NULL,
    poll INTEGER NOT NULL, FOREIGN KEY (poll)
REFERENCES polls(id),
    type INTEGER NOT NULL, FOREIGN KEY (type)
REFERENCES question_types(id)
);

CREATE TABLE metadata (
    id INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY
(id),
    question INTEGER, FOREIGN KEY (question) REFERENCES
questions(id),
    mkey VARCHAR(64) NOT NULL,
    mvalue VARCHAR(64) NOT NULL
);

CREATE TABLE variants (
    id INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY
(id),
    question INTEGER NOT NULL, FOREIGN KEY (question)
REFERENCES questions(id),
    vindex INTEGER NOT NULL,
    text VARCHAR(128) NOT NULL,
    link1 INTEGER, FOREIGN KEY (link1) REFERENCES
variants(id),
    link2 INTEGER, FOREIGN KEY (link2) REFERENCES
variants(id)
);

CREATE TABLE experts (
    id INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY
(id),
    username VARCHAR(64) UNIQUE NOT NULL,
    realname VARCHAR(32),
    surname VARCHAR(32),
    email VARCHAR(64) UNIQUE NOT NULL
);

CREATE TABLE sessions (
    id INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY
(id),
    timestamp TIMESTAMP NOT NULL,
    expert INTEGER NOT NULL, FOREIGN KEY (expert)
REFERENCES experts(id)
);

CREATE TABLE answers (
    id INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY
(id),
    session INTEGER NOT NULL, FOREIGN KEY (session)
REFERENCES sessions(id)
);

CREATE TABLE alternatives (
    id INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY
(id),
    note VARCHAR(256),
    answer INTEGER NOT NULL, FOREIGN KEY (answer)
REFERENCES answers(id),
    variant INTEGER NOT NULL, FOREIGN KEY (variant)
REFERENCES variants(id)
);

SELECT * FROM question_types;

```

```

# SELECT * FROM experts

# SELECT username FROM experts
# INNER JOIN sessions ON sessions.expert =
experts.id
# WHERE sessions.id = 1

package info.necauqua.dbosurvey;
import java.util.ArrayList;
import java.util.List;
public class Test {
    public static void main(String[] args) throws
InterruptedException {
        DBManager manager = new DBManager();

        Poll first = manager.getPoll(1);

        System.out.println(first.getQuestions().get(2).getTex
t());
        // manager.addPoll(createGamePoll());
        // manager.addPoll(createCafePoll());

        Thread.sleep(3000); // wait for it before
closing

        manager.close();
    }

    private static final QuestionType nOfMany = new
QuestionType("N of Many");
    private static final QuestionType matrix = new
QuestionType("Matrix");
    private static final QuestionType open = new
QuestionType("Open");
    private static final QuestionType sort = new
QuestionType("Sorted");

    private static Poll createGamePoll() {
        Poll poll = new Poll("Популярність ігор",
"bla-bla-bla, this is a test description");
        List<Question> qs = poll.getQuestions();

        Question q = new Question(poll, 1, "Як часто
Ви граєте в ігри?", nOfMany);
        qs.add(q);
        List<Metadata> md = q.getMetadata();
        List<Variant> vs = q.getVariants();

        md.add(new Metadata(q, "n_of_many.min",
"1"));
        md.add(new Metadata(q, "n_of_many.max",
"1"));
        vs.add(new Variant(q, 1, "Кожен день"));
        vs.add(new Variant(q, 2, "Декілька разів на
тиждень"));
        vs.add(new Variant(q, 3, "Декілька разів на
місяць"));
        vs.add(new Variant(q, 4, "Декілька разів на
рік"));
        q = new Question(poll, 2, "Якому виду ігор ви
надаєте перевагу?", nOfMany);
        qs.add(q);
        md = q.getMetadata();
        vs = q.getVariants();

        md.add(new Metadata(q, "n_of_many.min",
"1"));
        md.add(new Metadata(q, "n_of_many.max",
"*"));
        md.add(new Metadata(q, "open_answer.index",
"6"));
        vs.add(new Variant(q, 1, "Комп'ютерні"));
        vs.add(new Variant(q, 2, "Настільні"));
        vs.add(new Variant(q, 3, "Карткові"));
        vs.add(new Variant(q, 4, "Навчальні"));
        vs.add(new Variant(q, 5, "На відкритому
просторі"));
        vs.add(new Variant(q, 6, "Інші (відкрита
відповідь)"));

        q = new Question(poll, 3, "Ви граєте в
наступні комп'ютерні ігри?", matrix);
        qs.add(q);
        md = q.getMetadata();
        vs = q.getVariants();

        md.add(new Metadata(q,
"matrix.row_answers.min", "1"));
        md.add(new Metadata(q,
"matrix.row_answers.max", "1"));
        md.add(new Metadata(q, "matrix.rows", "10"));
        md.add(new Metadata(q, "matrix.columns",
"3"));
    }
}

```

```

List<Variant> rows = new ArrayList<>();
List<Variant> columns = new ArrayList<>();
rows.add(new Variant(q, 1, "Пригоди"));
rows.add(new Variant(q, 2, "Бойовики"));
rows.add(new Variant(q, 3, "Загадки"));
rows.add(new Variant(q, 4, "Логічні"));
rows.add(new Variant(q, 5, "MMORPG"));
rows.add(new Variant(q, 6, "RPG"));
rows.add(new Variant(q, 7, "Ігри-
симулятори"));
rows.add(new Variant(q, 8, "Спортивні"));
rows.add(new Variant(q, 9, "Стратегії"));
rows.add(new Variant(q, 10, "Гонки"));
columns.add(new Variant(q, 11, "Так"));
columns.add(new Variant(q, 12, "Час від
часу"));
columns.add(new Variant(q, 13, "Hi"));
addMatrix(q, vs, rows, columns);
q = new Question(poll, 4, "Як Ви зазвичай
отримуєте комп'ютерні ігри?", nOfMany);
qs.add(q);
md = q.getMetadata();
vs = q.getVariants();

md.add(new Metadata(q, "n_of_many.min",
"1"));
md.add(new Metadata(q, "n_of_many.max",
"1"));
vs.add(new Variant(q, 1, "Купую"));
vs.add(new Variant(q, 2, "Беру на прокат"));
vs.add(new Variant(q, 3, "Качаю з
інтернету"));
q = new Question(poll, 5, "Ви граєте на цих
пристроях?", nOfMany);
qs.add(q);
md = q.getMetadata();
vs = q.getVariants();

md.add(new Metadata(q, "n_of_many.min",
"1"));
md.add(new Metadata(q, "n_of_many.max",
"*"));
vs.add(new Variant(q, 1, "Комп'ютер"));
vs.add(new Variant(q, 2, "Ноутбук"));
vs.add(new Variant(q, 3, "Мобільний
телефон"));
vs.add(new Variant(q, 4, "Консоль"));
q = new Question(poll, 6, "Комп'ютер vs.
консоль?", nOfMany);
qs.add(q);
md = q.getMetadata();
vs = q.getVariants();

md.add(new Metadata(q, "n_of_many.min",
"1"));
md.add(new Metadata(q, "n_of_many.max",
"1"));
vs.add(new Variant(q, 1, "Комп'ютер"));
vs.add(new Variant(q, 2, "Консоль"));
vs.add(new Variant(q, 3, "Жоден з
варіантів"));
q = new Question(poll, 7, "PlayStation vs.
Xbox?", nOfMany);
qs.add(q);
md = q.getMetadata();
vs = q.getVariants();
md.add(new Metadata(q, "n_of_many.min",
"1"));
md.add(new Metadata(q, "n_of_many.max",
"1"));
vs.add(new Variant(q, 1, "PlayStation"));
vs.add(new Variant(q, 2, "Xbox"));
vs.add(new Variant(q, 3, "Жоден з
варіантів"));
q = new Question(poll, 8, "Назвіть свою
улюблену гру:", opel);
qs.add(q);

q = new Question(poll, 9, "Яку суму на рік Ви
витрачаєте на ігри?", nOfMany);
qs.add(q);
md = q.getMetadata();
vs = q.getVariants();

md.add(new Metadata(q, "n_of_many.min",
"1"));
md.add(new Metadata(q, "n_of_many.max",
"1"));
vs.add(new Variant(q, 1, "0 - 500 грн"));
vs.add(new Variant(q, 2, "501 - 1000 грн"));
vs.add(new Variant(q, 3, "1 001 - 2 500
грн"));
vs.add(new Variant(q, 4, "Більше"));

```

```

q = new Question(poll, 10, "Вкажіть вашу
стать:", nOfMany);
qs.add(q);
md = q.getMetadata();
vs = q.getVariants();
md.add(new Metadata(q, "n_of_many.min",
"1"));
md.add(new Metadata(q, "n_of_many.max",
"1"));
vs.add(new Variant(q, 1, "Чоловіча"));
vs.add(new Variant(q, 2, "Жіноча"));
return poll;
}

private static Poll createCafePoll() {
    Poll poll = new Poll("Оцінка кафе", "bla-bla-
bla, this is a test description");
    List<Question> qs = poll.getQuestions();

    Question q = new Question(poll, 1, "Як часто
ви відвідуєте кафе?", nOfMany);
    qs.add(q);
    List<Metadata> md = q.getMetadata();
    List<Variant> vs = q.getVariants();

    md.add(new Metadata(q, "n_of_many.min",
"1"));
    md.add(new Metadata(q, "n_of_many.max",
"1"));
    vs.add(new Variant(q, 1, "Щодня"));
    vs.add(new Variant(q, 2, "Декілька разів на
тиждень"));
    vs.add(new Variant(q, 3, "Раз в тиждень"));
    vs.add(new Variant(q, 4, "Декілька разів на
місяць"));
    vs.add(new Variant(q, 5, "Раз в місяць"));
    vs.add(new Variant(q, 6, "Декілька разів на
рік"));
    vs.add(new Variant(q, 7, "Вперше"));
    q = new Question(poll, 2, "Зазначте настільки
ви згодні з наступними висловлюваннями?", matrix);
    qs.add(q);
    md = q.getMetadata();
    vs = q.getVariants();
    md.add(new Metadata(q,
"matrix.row_answers.min", "1"));
    md.add(new Metadata(q,
"matrix.row_answers.max", "1"));
    md.add(new Metadata(q, "matrix.rows", "5"));
    md.add(new Metadata(q, "matrix.columns",
"4"));

    List<Variant> rows = new ArrayList<>();
    List<Variant> columns = new ArrayList<>();

    rows.add(new Variant(q, 1, "Закуси були
поданні відмінно"));
    rows.add(new Variant(q, 2, "Закуси були
якісні і смачні"));
    rows.add(new Variant(q, 3, "Ціна була
приваблива"));
    rows.add(new Variant(q, 4, "Інтер'єр кафе був
приємним"));
    rows.add(new Variant(q, 5, "WiFi підключення
достатньо швидке і якісне"));

    columns.add(new Variant(q, 6, "Повністю
погоджуюсь"));
    columns.add(new Variant(q, 7, "Погоджуюсь"));
    columns.add(new Variant(q, 8,
"Непогоджуюсь"));
    columns.add(new Variant(q, 9, "Зовсім
непогоджуюсь"));

    addMatrix(q, vs, rows, columns);
    q = new Question(poll, 3, "Відсортуйте від
найкращого до найгіршого", sort);
    qs.add(q);
    vs = q.getVariants();

    List<Variant> toSort = new ArrayList<>();
    toSort.add(new Variant(q, 1, "Інтер'єр"));
    toSort.add(new Variant(q, 2,
"Обслуговування"));
    toSort.add(new Variant(q, 3, "Смак страв"));
    toSort.add(new Variant(q, 4, "Смак напоїв"));
    toSort.add(new Variant(q, 5, "Чистота у
вбиральні"));
    toSort.add(new Variant(q, 6, "Місце
розташування закладу"));
    addMatrix(q, vs, toSort, toSort);
    // addMatrix(q, vs, toSort, toSort.stream()
    // .map(v -> new

```

```

Variant(v.getQuestion(), v.getIndex(), ""))
//                                .collect(Collectors.toList()));

    q = new Question(poll, 4, "Оцініть наше
кафе", nOfMany);
    qs.add(q);
    md = q.getMetadata();
    vs = q.getVariants();

    md.add(new Metadata(q, "n_of_many.min",
"1"));
    md.add(new Metadata(q, "n_of_many.max",
"1"));

    vs.add(new Variant(q, 1, "1"));
    vs.add(new Variant(q, 2, "2"));
    vs.add(new Variant(q, 3, "3"));
    vs.add(new Variant(q, 4, "4"));
    vs.add(new Variant(q, 5, "5"));

    q = new Question(poll, 5, "Чи порекомендували
б ви наше кафе?", nOfMany);
    qs.add(q);
    md = q.getMetadata();
    vs = q.getVariants();

    md.add(new Metadata(q, "n_of_many.min",
"1"));
    md.add(new Metadata(q, "n_of_many.max",
"1"));

    vs.add(new Variant(q, 1, "Так"));
    vs.add(new Variant(q, 2, "Ні"));
    q = new Question(poll, 6, "Вкажіть вашу
стать:", nOfMany);
    qs.add(q);
    md = q.getMetadata();
    vs = q.getVariants();

    md.add(new Metadata(q, "n_of_many.min",
"1"));
    md.add(new Metadata(q, "n_of_many.max",
"1"));

    vs.add(new Variant(q, 1, "Чоловіча"));
    vs.add(new Variant(q, 2, "Жіноча"));

    q = new Question(poll, 7, "Вкажіть вашу
вікову групу:", nOfMany);
    qs.add(q);
    md = q.getMetadata();
    vs = q.getVariants();

    md.add(new Metadata(q, "n_of_many.min",
"1"));
    md.add(new Metadata(q, "n_of_many.max",
"1"));

    vs.add(new Variant(q, 1, "<20"));
    vs.add(new Variant(q, 2, "21-30"));
    vs.add(new Variant(q, 3, "31-40"));
    vs.add(new Variant(q, 4, "41-50"));
    vs.add(new Variant(q, 5, "51-60"));
    vs.add(new Variant(q, 6, "60+"));

    return poll;
}

private static void addMatrix(Question q,
List<Variant> vs, List<Variant> rows, List<Variant>
columns) {
    vs.addAll(rows);
    if(columns != rows) {
        vs.addAll(columns);
    }
    for(Variant row : rows) {
        for(Variant column : columns) {
            Variant v = new Variant(q, 0, "");
            row.setLink1(v);
            column.setLink1(v);
            v.setLink1(row);
            v.setLink2(column);
            vs.add(v);
        }
    }
}

package info.necauqua.dbosurvey;
package info.necauqua.dbosurvey;

import org.hibernate.HibernateException;

```

```

import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import org.hibernate.cfg.Configuration;

public class DBManager {

    private final SessionFactory sessionFactory;

    public DBManager() {
        try {
            sessionFactory = new Configuration()

.addPackage("info.necauqua.dbosurvey")
.addAnnotatedClass(Alternative.class)
.addAnnotatedClass(Answer.class)
.addAnnotatedClass(Expert.class)
.addAnnotatedClass(Metadata.class)
.addAnnotatedClass(Poll.class)
.addAnnotatedClass(Question.class)

.addAnnotatedClass(QuestionType.class)
.addAnnotatedClass(Session.class)
.addAnnotatedClass(Variant.class)
.configure()
.buildSessionFactory();
        } catch (Throwable ex) {
            throw new
ExceptionInInitializerError(ex);
        }
    }

    public void close() {
        sessionFactory.getCurrentSession().close();
        sessionFactory.close();
    }

    public void addPoll(Poll poll) {
        Transaction tx = null;
        org.hibernate.Session session =
sessionFactory.getCurrentSession();
        try {
            tx = session.beginTransaction();
            session.save(poll);
            for(Question question :
poll.getQuestions()) {
                session.saveOrUpdate(question.getType());
                session.save(question);
                for(Variant variant :
question.getVariants()) {
                    Variant link1 =
variant.getLink1();
                    if(link1 != null) {
                        session.save(link1);
                    }
                    Variant link2 =
variant.getLink2();
                    if(link2 != null) {
                        session.save(link2);
                    }
                    session.save(variant);
                }
                for(Metadata metadata :
question.getMetadata()) {
                    session.save(metadata);
                }
            }
            tx.commit();
        } catch (HibernateException e) {
            if(tx != null) {
                tx.rollback();
            }
            e.printStackTrace();
        }
    }

    public Poll getPoll(int id) {
        Transaction tx = null;
        org.hibernate.Session session =
sessionFactory.getCurrentSession();
        try {
            tx = session.beginTransaction();
            return session.get(Poll.class, id);
        } catch (Exception e) {
            if(tx != null) {
                tx.rollback();
            }
            e.printStackTrace();
            return null;
        }
    }
}

```

