ДИПЛОМНА РОБОТА
БАКАЛАВРА

ДРБ. ІП - 38.00.00.000 ПЗ

Група ІП-18-2

Споденюк Андрій

2022

Міністерство освіти і науки України
Івано-Франківський національний технічний університет нафти і газу

Інститут інформаційних технологій

Кафедра інженерії програмного забезпечення

УДК 004.4

ДИПЛОМНА РОБОТА БАКАЛАВРА

Тема: Інформаційні моделі візуалізації процесів прийняття рішень

(назва згідно з наказом ректора)

Спеціальність – 121 "Інженерія програмного забезпечення"

ПОЯСНЮВАЛЬНА ЗАПИСКА

ДРБ.ІП - 38.00.00.000 ПЗ

(позначення)

Студент

ІП-18-2 /А.В.Споденюк/
(шифр групи) (підпис) (дата) (розшифровка підпису)

Керівник роботи

асист. /Л.М. Гобир/
(посада) (підпис) (дата) (розшифровка підпису)

Консультанти:

з питань нормоконтролю

доц., к.т.н /В.Я. Піх/
(посада) (підпис) (дата) (розшифровка підпису)

Допускається до захисту
Завідувач кафедри

проф., д.т.н. /В.І.Шекета/
(посада) (підпис) (дата) (розшифровка підпису)

2022
(рік)

Івано-Франківський національний технічний університет нафти і газу

Інститут, факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Ступінь вищої освіти бакалавр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:
Зав. кафедрою **ІПЗ**
проф. **В.І. Шекета**
“ ____ ” _____2022 р.

ЗАВДАННЯ

НА ДИПЛОМНУ РОБОТУ БАКАЛАВРА СТУДЕНТОВІ

Споденюкові Андрію Вікторовичу

(прізвище, ім’я, по-батькові)

1. Тема проекту (роботи) Інформаційні моделі візуалізації процесів прийняття рішень
керівник проекту (роботи) Гобир Лідія Мирославівна асистент

затверджені наказом закладу вищої освіти від **“** _ __ ” __ 2022 р. № _____

2. Строк подання студентом проекту (роботи)

3.Вихідні дані до проекту (роботи)

4. Зміст розрахунково - пояснювальної записки (перелік питань, які потрібно розробити)

1. Вступ до проблеми
2. Огляд існуючих концепцій, рішень та сервісів в даній області
3. Побудова моделі або алгоритму власного рішення
4. Документування реалізації власного оригінального рішення вибраними засобами

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

1. Перехідні стани моделі WBM (рис. 1.1)
2. Діаграми архітектури JADE. (рис. 1.5)
3. Алгоритм роботи фреймворка (рис 2.3)
4. Коефіцієнт адаптивності запиту (рис 3.2)
5. Загальна задоволеність користувачів системою для різних груп (рис. 3.6)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання **25 травня 2022 р.**

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту	Примітка
1	Визначення та обгрунтування теми роботи	25.05.2022	виконано
2	Огляд існуючих концепцій, рішень та сервісів в даній області	26.05.2022	виконано
3	Побудова моделі або алгоритму власного рішення	29.05.2022	виконано
4	Документування реалізації власного оригінального продукту підтримки прийняття рішення вибраними засобами	10.06.2022	виконано
5	Оформлення пояснювальної записки кваліфікаційної роботи	14.06.2022	виконано

Студент _____
(підпис)

Керівник роботи _____
(підпис)

АНОТАЦІЯ

В дипломній роботі пропонується підхід та структура для вимірювання досвіду та результативності користувачів системи підтримки прийняття рішень у сфері охорони здоров'я. Основна увага зосереджена на виборі візуалізацій користувачами на основі характеру створених запитів. Метою цієї структури є збір індивідуальних даних користувацького досвіду і фактичних результатів, щоб отримати профіль користувачів для подальших досліджень. Модель, що використовується, є схемою зважування, розробленою для підтримки подальших розширень і вдосконалень за допомогою «техніки міркування ШІ». Результати цього дослідження можуть бути використані для підвищення точності та скорочення часу вибору візуалізації через багаторазове використання, і тому важливим як підхід до підвищення продуктивності як такий.

КЛЮЧОВІ СЛОВА: ПІДТРИМКА ПРИЙНЯТТЯ РІШЕНЬ, ВІЗУАЛІЗАЦІЯ, МОДЕЛЬ, ПРОДУКТИВНІСТЬ, АДАПТИВНІСТЬ, ПРАВИЛА, МІРКУВАННЯ

ANNOTATION

The thesis proposes an approach and framework for measuring the experience and performance of users of a decision support system in healthcare. The main focus is on the choice of visualizations by users based on the nature of the generated queries. The purpose of this framework is to collect individual user experience data and actual results to obtain a user profile for further research. The model used is a weighting scheme designed to support further extensions and improvements using "AI reasoning techniques". The results of this study can be used to improve the accuracy and reduce the time of visualization selection through repeated use, and therefore important as a productivity improvement approach as such.

KEY WORDS: DECISION SUPPORT, VISUALIZATION, MODEL, PRODUCTIVITY, ADAPTIVENESS, RULES, REASONING

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ……..7

ВСТУП……..……8

1. **ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ**

1.1. Основи адаптації……10

1.2. Системи підтримки прийняття рішень……..…..14

1.3. Аналіз відомих джерел предметної області ….……..…..22

1.4. Візуалізація даних……..25

2. **ПРОПОНУВАНА АДАПТИВНА ОСНОВА ВІЗУАЛІЗАЦІЇ ДАНИХ**

2.1. Архітектура системи адаптивної візуалізації даних……...33

2.2. Архітектура системи……..35

2.3. Багатомовне розгортання……..38

2.4. Реалізація……42

2.5. Дослідження юзабіліті……...45

2.6. Пропонований метод тестування юзабіліті…….49

3. **ВЛАСНЕ ОРИГІНАЛЬНЕ АЛГОРИТМІЧНЕ ТА ПРОГРАМНЕ РІШЕННЯ ЗАДАЧІ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ**

3.1. Складові програмного рішення……51

3.2. Опис фреймворку для користувачів……52

3.3. Завдання етапу проектування……...53

3.4. Час проектування рішення……55

3.5. Адаптація……57

ВИСНОВКИ……61

СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА……64

БІБЛІОГРАФІЧНА ДОВІДКА

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ППР – підтримка прийняття рішень

БД – База даних

СУБД - Система управління базами даних

ПЗ - Програмне забезпечення

ІС – інформаційна система

ІА – інтелектуальний аналіз

ВСТУП

Адаптивні підходи використовуються для покращення користувацького досвіду та покращення якості перегляду веб-сторінок на основі інформації профілю, зібраної в результаті взаємодії користувачів. У системах підтримки прийняття рішень зросла потреба в адаптації персоналізації, щоб надати більше негайної та актуальної інформації особам, які приймають рішення, використовуючи веб-доступ до даних. Використання візуалізацій для відтворення складних результатів запитів у реальному часі має особливе значення в багатьох областях.

Інтернет - це середовище, яке постійно розвивається. Інформаційні системи надають зростаючу кількість оброблених даних як професійним користувачам, так і великій кількості звичайних користувачів по всьому світу. Раніше веб-сайти та інформаційні системи надавали користувачам дуже повільні та статичні дані, але сьогодні веб-додатки стають дедалі складнішими, забезпечуючи користувачам надійні та динамічні дані. Розробляються прості у використанні мови інтерфейсу та запитів, які є частиною інформаційних систем і передають і відображають дані у візуалізацію, щоб зрозуміти природу даних , яка визначається як природа наборів даних, таких як булеві набори даних, цілочисельні набори даних тощо. Дані доступні користувачам у різних складних формах, які стосуються фактів, подій і транзакцій у необробленому вигляді, коли ці дані обробляються таким чином, щоб мати значення для клієнта, який їх отримує. Дослідження показали, що існує величезна кількість доступних даних, але вони не завжди використовуються для надання точної та необхідної інформації. Для ефективної доставки даних представлені різні архітектури. Дані, доставлені після кількох запитів, не мають користі, якщо користувачі, які аналізують дані, не можуть проаналізувати дані належним чином. Щоб подолати це, використовуються різні фреймворки для доставки оброблених даних запиту у вигляді візуалізації. Одним з цих типів підходів є «персоналізація інформації». Посилаючись тут на персоналізацію інформації, ми маємо на увазі надання

персоналізованих даних кінцевим користувачам відповідно до вимог і конкретних потреб і ролей. Ці типи інформації розумно доставляють користувачам, які не знають про різні архітектурні та поведінкові аспекти компонентів.

Адаптивний Інтернет — це єдина область, яка спрямована на подолання розриву між людськими емоціями та обчислювальними технологіями. Адаптивні веб-додатки — це інтелектуальні програми, які надають потрібний контент користувачам у потрібний час, враховуючи їхні уподобання. Кожен користувач може по-різному дивитися на ту саму проблему. Адаптивні веб-техніки починаються на тому ж рівні, і коли користувачі взаємодіють із додатками, вони продовжують відстежувати їхні емоції, симпатії чи антипатії, їхні уподобання, їхні оцінки, а також їхні наближення. Концептуально і практично впроваджено цілий ряд прийомів адаптації.

Ми пропонуємо підхід та структуру для вимірювання досвіду та компетенцій окремих користувачів системи підтримки прийняття рішень у сфері охорони здоров’я. Основна увага зосереджена на виборі візуалізацій користувачами на основі оцінки характеру створених запитів. Структура спрямована на збір індивідуального досвіду та рівня адаптивності користувачів, щоб отримати профіль користувачів для подальших досліджень. Модель, що використовується, є схемою зважування, але розроблена для підтримки подальших розширень і вдосконалень з використанням методів міркувань на основі ШІ. Система підтримує підвищену точність і скорочення часу вибору візуалізації в порівнянні з повторним використанням, і тому є важливою як підхід до підвищення продуктивності.

1 ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ТА СЕРВІСІВ В ПРЕДМЕТНІЙ ОБЛАСТІ

1.1 Основи адаптації

Модель веб-поведінки (WBM)

Модель WBM – це автомат із тимчасовим переходом стану для представлення класів поведінки користувачів у мережі [3]. Графічно моделі WBM виражаються міченими графіками, що дозволяє легко зрозуміти синтаксис. В основному, існують різні типи станів, які завантажуються у веб-браузери та спеціальні скрипти, тобто WBM-скрипт. Цей скрипт запускається, щоб знати про поточний стан і в міру його зміни; його приймає інша держава. Отже, існує принаймні один початковий стан, а також один стан, що приймає. Ця модель не охоплює всі альтернативні навігації.

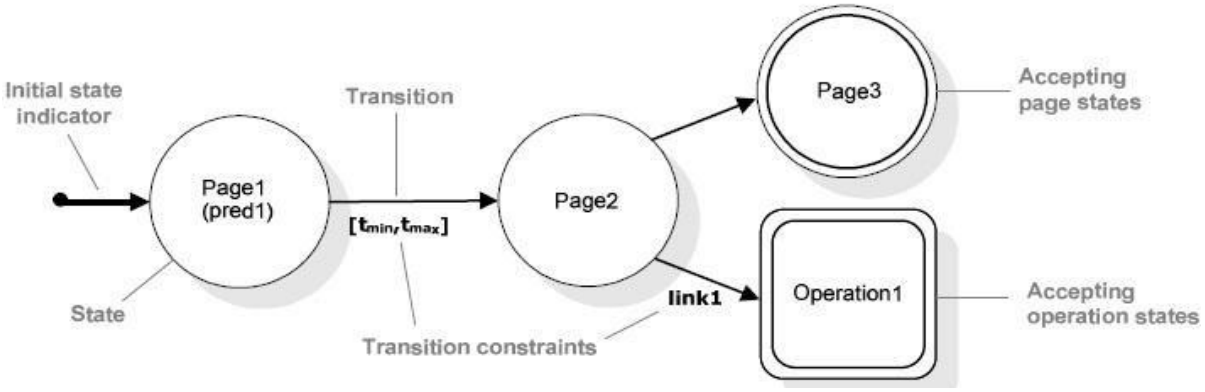


Рисунок 1.1 Перехідні стани моделі WBM

Рисунок описує перехідні стани, визначені в WBM з різними переходами. Обмеження стану – введення стану може бути предметом оцінки обмеження стану, що виражає предикат властивостей сторінок, до яких здійснюється доступ, або операції, що запускається [3]. Такий предикат може посилатися на вміст, що

відображається на сторінках, або на параметри роботи. Доступ до стану здійснюється лише в тому випадку, якщо оцінка предиката дає істину [3].

Обмеження посилань – кожен перехід може бути позначений назвою посилання, що входить на сторінку або дозволяє операцію. Доступ до стану здійснюється лише за умови навігації за вказаним посиланням. Часові обмеження – кожен перехід від вихідного до цільового стану може бути позначений парою [tmin, tmax], що виражає інтервал часу, протягом якого може відбутися перехід. tmin або tmax можуть бути відсутніми, що вказує на відкриті межі інтервалу. Якщо перехід не запускається протягом tmax одиниць часу, він більше не може відбутися; з іншого боку, навігаційні дії, які відбуваються до tmin, втрачаються [3]. Таким чином, використання відповідних часових обмежень може призвести до скасування запущених сценаріїв. Одним із важливих аспектів моделей WBM є те, що не всі альтернативи навігації повинні бути охоплені. Оскільки мета WBM полягає в тому, щоб охопити стислий набір взаємодій користувачів, описуючи конкретні цілі навігації та відповідні «віхи»; релевантною є лише підмножина всіх можливих альтернатив навігації. Веб-сайти електронної комерції, наприклад, інтенсивно використовують так звані сторінки доступу, які служать лише цілям надання користувачам доступних для перегляду категорій для отримання фактично пропонованих продуктів [3]. Крім того, веб-сайти зазвичай надають кілька різних шляхів доступу до свого основного вмісту. Тому, зосереджуючись лише на взаємодіях, які дійсно виражають цілі навігації, WBM дозволяє як абстрагуватися від непотрібних деталей, так і визначати невеликі й легко зрозумілі специфікації. Таким чином, зміна стану WBM може спричинити лише виконання визначених цільових взаємодій – у змодельованому порядку.

Обмеження моделі WBM

WBM — це автомат із тимчасовим переходом стану для представлення класів поведінки користувачів у мережі. Обмеження моделі WBM полягає в тому, що шаблони адаптації вже розроблені на етапі проектування, не вимагаючи

глибокого знання фактичної структури програми. Для складних шаблонів адаптації системи немає способу розробити коефіцієнт адаптації на стадії проектування, тому це спонукає нас створити адаптивну модель, в якій на стадії проектування не визначено жодних фіксованих правил і яка може змінювати коефіцієнт адаптації відповідно до введених користувачів.

Модель правил ECA (ECA)

У розвитку верхньої моделі вводиться нова модель правил ECA, яка поєднує сценарії WBM і систему адаптації Webml. У цій моделі вводиться механізм правил, що містить певний набір правил, і коли стан змінюється, визначається нове правило і створюється новий сценарій, який переміщується від однієї URL-адреси до іншої. Механізм правил ділиться даними та зберігає дані про завершені стани з профілями користувачів. Механізм правил не прив'язаний до використовуваної технології веб-додатків і також може працювати з різними веб-додатками. У разі збою механізму правил веб-додатки працюють, і механізм правил можна відновити незалежно [3]. Тому механізм правил фіксує записи переходу та зберігає їх для створення нових правил.

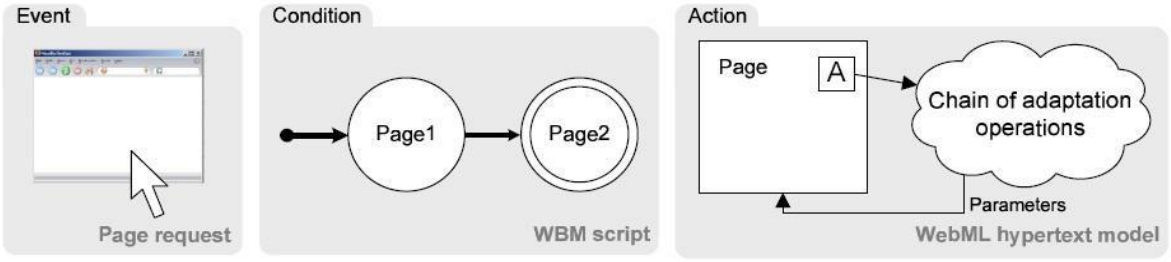


Рисунок 1.2 - Модель правил ECA

На рисунку показана модель правил ECA, яка діє у відповідь на відвідування користувачами сторінки 1 з наступним відвідуванням сторінки 2. Таким чином, виражена умова залишається істинною лише тоді, коли сценарій потрапляє в стан прийняття Page2. Після досягнення стану прийняття, дії (виражені як хмара

на рисунку 1.2) виконуються, і після повторного обчислення параметрів сторінки можуть бути виконані можливі адаптації.

Обмеження моделі правила ECA

Модель правил ECA була введена з механізмом правил адаптації, який фіксує поведінку і пізніше їх можна використовувати для визначення адаптивного фактора. Але це працює лише в умовах «включено та якщо». Якщо відбудеться конкретна подія, то будуть показані лише адаптивні результати. Також це адаптивне правило суворе до певної сторінки, яку переглядає користувач. Тому нам потрібен механізм правил адаптації, в якому будуть фіксуватися всі події і який пізніше можна використовувати для надання адаптивних результатів, коли відбувається певна подія. Механізм правил адаптації також можна використовувати для охоплення всіх видів поведінки, а пізніше ці результати можна використовувати для підвищення ефективності системи.

Модель адаптації для людей з вадами зору

Це структура, яка дозволяє людям з вадами зору отримати доступ до графіки в Інтернеті. Він обмежується певним типом користувачів та обладнанням, але тепер введено новий фреймворк, який працює адаптивно з урахуванням різних профілів користувачів [4].

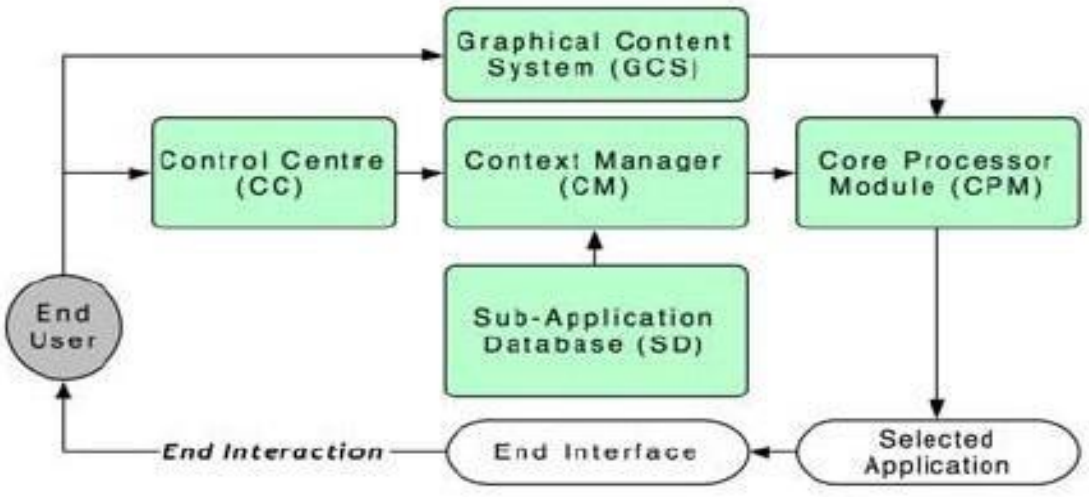


Рисунок 1.3. Компоненти системи адаптації людей з вадами зору

До основних компонентів системи відносяться:

а) База даних підпрограм (SD): - це окрема програма, яка здатна або витягувати інформацію з векторної графіки, або відображати растрову графіку. Він обробляє зібрані дані про графіку та перетворює їх у доступний печінковий, тактильний, аудіо (або комбінацію будь-якого) інтерфейсу [4]. Існує кілька причин, чому ці функції включені в різні незалежні підпрограми.

b) Система графічного вмісту (GCS): GCS надає інтерфейс для запуску додатку зі сприйняттям інформації про графічний вміст, обраний кінцевим користувачем. Він також визначає тип і формат графіки після того, як користувач виконує дію тригера. Також GCS відповідає за надсилання інформації до CPM для адаптації контенту.

c) Центр керування (CC):- Центр керування дозволяє користувачам читати та змінювати свої профілі та налаштування, конфігурації системи та налаштування підпрограми. Хоча профілі користувачів створюються під час процесу встановлення системи, користувач все одно може змінити свої налаштування вручну пізніше. Він також забезпечує функції імпорту та експорту.

d) Менеджер контексту (CM):- Він зберігає та керує контекстною інформацією, тобто інформацією, необхідною для адаптації, яка включає інформацію про

а) графічний вміст,

б) конфігурацію системи та апаратні/програмні компоненти допоміжних технологій,

в) вимоги (наприклад, конфігурація системи) і зворотний зв'язок, наданий кожною піддодатком (обидві дані вказуються розробниками піддодатків),

г) профіль користувача та його налаштування [4].

e) Модуль основного процесора (CPM):- CPM виконує адаптивну логіку, збираючи вхідні дані, надані як GCS, так і CM. CPM виконує адаптацію, щоб

знайти найбільш адекватний кінець інтерфейс для своїх користувачів [4]. Він також спілкується з доступними допоміжними технологіями.

Обмеження адаптивної порушеної моделі

Ця модель розроблена спеціально для людей з вадами зору. Ця модель вимагає різного роду візуалізацій та обладнання, щоб фактична система працювала. Основним обмеженням цієї моделі є те, що людям потрібно змінити апаратне забезпечення, щоб фактична система була адаптована. Але в той же час, як і в адаптивній моделі, представленій тут, користувачі мають різні види візуалізації, і користувачі повинні вибрати найбільш підходящу візуалізацію, щоб вони могли зрозуміти та отримати доступ до інших візуалізацій.

1.2 Системи підтримки прийняття рішень

Прийняття рішень, що стосуються складних систем (наприклад, управління організаційними операціями, промисловими процесами або інвестиційними портфелями, командування військовими частинами або контроль атомних електростанцій) часто напружує наші когнітивні здібності [6]. Незважаючи на те, що індивідуальні взаємодії між змінними системи можна добре зрозуміти, передбачити, як система відреагує на зовнішню маніпуляцію, як-от політичне рішення, часто важко. Існує значна кількість емпіричних доказів того, що людські інтуїтивні судження та прийняття рішень можуть бути далекі від оптимальних, і вони ще більше погіршуються зі складністю та стресом. Оскільки в багатьох ситуаціях якість рішень має важливе значення, допомога у вирішенні недоліків людських суджень і прийняття рішень була основним фокусом науки протягом усієї історії [7]. Такі дисципліни, як статистика, економіка та дослідження операцій, розробили різні методи раціонального вибору. Ці методи, часто доповнені різноманітними методами, що походять із інформатики, когнітивної психології та штучного інтелекту, були реалізовані у вигляді комп’ютерних програм або як самостійні інструменти, або як інтегровані

обчислювальні середовища для прийняття складних рішень. Таким середовищам часто дають загальну назву систем підтримки прийняття рішень (DSS).

Системи підтримки прийняття рішень набули популярності насамперед через їхню здатність задовольняти потребу прийняття рішень наданою інформацією. Система підтримки прийняття рішень — це клас інформаційних систем, які підтримують діяльність бізнесу та організації прийняття рішень. DSS поєднує інтелектуальні ресурси індивідів із можливостями комп'ютера для підвищення якості рішень [7]. Це комп’ютерна підтримка для тих, хто приймає управлінські рішення, які мають справу з напівструктурованими проблемами. Правильно розроблена DSS – це інтерактивна програмна система, призначена для того, щоб допомогти особам, які приймають рішення, зібрати корисну інформацію з комбінації необроблених даних, документів, особистих знань або бізнес-моделей для виявлення та вирішення проблем і прийняття рішень.

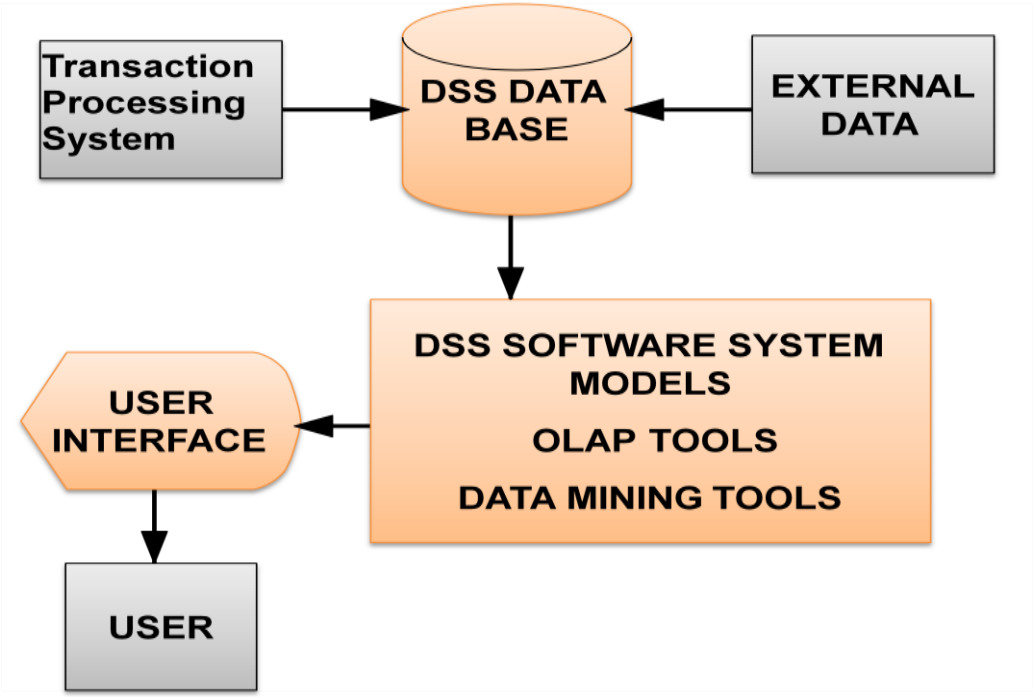


Рисунок 1.4 - Потік системи підтримки прийняття рішень

Система підтримки прийняття рішень – це загальний термін для будь-якого комп’ютерного додатка, який покращує здатність людини або групи приймати

рішення. Він також може використовуватися як інструмент, у який користувач вводить дані, а програмний компонент обробляє дані, а рішення приймаються на основі наданої інформації. Для того, щоб зробити інструмент прийняття рішень, усі основні компоненти системи повинні бути розглянуті в системі, щоб отримати оптимальні результати.

Розглянемо приклад: національний продавець книг в Інтернеті хоче почати продавати свою продукцію на міжнародному рівні, але спочатку має визначити, чи буде це розумним бізнес-рішенням. Постачальник може використовувати DSS для збору інформації зі своїх власних ресурсів (за допомогою такого інструменту, як OLAP), щоб визначити, чи має компанія здатність або потенційну здатність розширити свій бізнес, а також із зовнішніх ресурсів, таких як галузеві дані, щоб визначити, чи справді є попит на задоволення. DSS збере та проаналізує дані, а потім представить їх у спосіб, який може бути інтерпретований людьми. Деякі системи підтримки прийняття рішень дуже близькі до того, щоб діяти як агенти штучного інтелекту. Програми DSS - це не окремі інформаційні ресурси, такі як база даних або програма, яка графічно відображає показники продажів, а комбінація інтегрованих ресурсів, що працюють разом.

Системи підтримки прийняття рішень є цінними в ситуаціях, коли обсяг доступної інформації є непосильним для інтуїції людини, яка приймає рішення без сторонньої допомоги, і в яких точність і оптимальність мають значення [8]. Системи підтримки прийняття рішень можуть допомогти людським когнітивним недолікам, інтегруючи різні джерела інформації, забезпечуючи розумний доступ до відповідних знань і допомагаючи процесу структурування рішень. Вони також можуть підтримувати вибір серед чітко визначених альтернатив і спиратися на формальні підходи, такі як методи інженерної економіки, дослідження операцій, статистика та теорія рішень. Вони також можуть використовувати методи штучного інтелекту для вирішення евристичних проблем, які неможливо розв’язати формальними методами. Правильне

застосування інструментів прийняття рішень підвищує продуктивність, ефективність та результативність і дає багатьом підприємствам порівняльну перевагу перед конкурентами, дозволяючи їм робити оптимальний вибір технологічних процесів та їх параметрів, планувати бізнес-операції, логістику чи інвестиції.

Прийняття рішень є фундаментальною управлінською діяльністю. Його можна концептуалізувати як таке, що складається з чотирьох етапів: розвідка, проектування, вибір і впровадження.

Важливі аспекти системи підтримки прийняття рішень

Нижче ми перерахуємо ті аспекти, які вважаються важливими для систем підтримки прийняття рішень.

Найважливішим фактором є простота використання системи підтримки прийняття рішень – її здатність дозволяти людям, які не мають технічних знань, мати справу з нею безпосередньо. Найбільшою і найтривкішою проблемою комп'ютерів була їхня гнучкість, нездатність дозволити людині, яка насправді потребує даних, мати справу безпосередньо з комп'ютером.

Можливість доступу до інформації не повинна обмежуватися лише частиною організації або лише певними управлінськими або професійними групами. Замість цього ресурс повинен бути розподілений серед усіх людей і частини організації, які його потребують, без широкого доступу; Потужність передової системи розподіленої обробки залишиться невикористаною, як це зазвичай було в минулому.

Ідеальна система підтримки прийняття рішень, на відміну від попереднього методу розробки додатків, не повинна бути «системою» у строгому сенсі цього терміну. Скоріше, це має бути високоадаптивний генератор підтримки прийняття рішень, який професіонали можуть легко використовувати для швидкого розробки прототипів підтримки даних, які підходять для кожного конкретного завдання прийняття рішень. Цей адаптивний інструмент повинен

дозволяти швидкі зміни дизайну, якщо оригінальний дизайн не відповідає стилю або потребам людини в збиранні інформації.

Щоб адекватно підтримувати людський елемент, ця високоадаптивна підтримка повинна мати можливість надавати доступ до оперативних даних, а також до підсумкових даних, які вже були оброблені прикладними програмами, розробленими для інших конкретних операційних завдань. Не менш важливо, що цей інструмент повинен надавати фахівцям доступ до вихідних даних організації, а також дозволяти отримати доступ за один крок за допомогою однієї нескладної процедури або команди без необхідності повторно вводити неузагальнені дані.

Організаціям іноді потрібен доступ до вихідних даних, оскільки ефективність пов'язана з тим, наскільки добре вихідні дані організовані в системі; тому генератор підтримки прийняття рішень повинен мати можливість взаємодіяти з справжньою СУБД. Він також повинен мати можливість отримати доступ до стандартних «плоских» файлів опосередковано, використовуючи потужність хост-комп'ютера, щоб полегшити як інтерфейс користувача, так і доступ до даних без зміни існуючих файлів.

Керівна або професійна інформаційна робоча станція включатиме клавіатуру, екран і інтерфейс до принтера, який може друкувати все, від прямого тексту до графіки, як-от кругові діаграми, стовпчасті діаграми та лінійні діаграми.

Інструмент підтримки повинен взаємодіяти з кількома різними системами і можливостями, він повинен бути сумісним з усіма з них, інструмент повинен надавати користувачам єдину мову, яка легко використовується для доступу до маніпулювання та представлення даних у спосіб, який найкраще підтримуватиме кінцевий результат. користувач.

Можливості, які пропонує DSS:

1) Підтримка прийняття рішень у погано структурованих ситуаціях – у яких саме через відсутність структури проблеми не піддаються повній

комп’ютеризації, але потребують комп’ютерної допомоги для доступу та обробки великої кількості даних.

2) Допоможіть швидко отримати результати, необхідні для прийняття рішення.

3) Підтримка різних етапів процесу прийняття рішень.

4) Сприяти прийняттю високоякісних рішень, заохочуючи прийняття рішень на основі інтеграції доступної інформації та людських суджень.

5) Запропонуйте гнучкість на відміну від заздалегідь визначеного зразка використання, що дозволяє легко пристосуватися до певного стилю прийняття рішень.

6) Сприяти виконанню рішень, які часто виходять за межі відділу.

7) Підтримка прийняття групових рішень, зокрема через групову DSS (GDSS).

8) Дайте організації можливість краще зрозуміти свій бізнес шляхом розробки та роботи з моделями.

Системи підтримки прийняття рішень використовуються в різних галузях від військової, освітньої та будь-якої іншої галузі. Інструменти DSS використовуються для підвищення ефективності системи, а не ефективності систем. Ці інструменти є потенційними інструментами вимірювання, які використовуються для цілей управління. У DSS рішення значною мірою залежать від якості інформації. Під час прийняття будь-яких рішень найкращі сценарії завжди розглядаються без збоїв у системі. Проблеми виникають, коли кількість доступної інформації величезна і неоднорідна, а її якість неможливо визначити заздалегідь. Системи підтримки прийняття рішень для спостереження за складною системою є складним завданням для створення. Складні системи мають високу динамічність за своєю природою, які складаються з різних параметрів, які мають специфічний вплив на систему. Коли ми маємо всі важливі параметри складної системи в СППР, легко зробити висновок про рішення, але важко вирішити, який параметр має більший вплив на систему, ніж інший.

У розділі проаналізували модель правил адаптації WBM, яка визначає правила адаптації, за допомогою яких можна побудувати адаптовані системи, щоб надати користувачам адаптовані результати. Вона має обмеження, що правила адаптації вже розроблені в системі. На основі нашого подальшого аналізу ми виявили, що просування адаптації полягає в створенні механізму правил, у якому визначені певні правила, і ці правила можна використовувати для подальших адаптаційних факторів. Основним обмеженням є те, що він працює лише з «увімкнено та якщо ” умови. Якщо ці умови можна усунути, то цей механізм правила може бути використаний для надання фактичних адаптаційних факторів. Потім ми вивчали адаптивний механізм, який використовується для людей з вадами зору, щоб забезпечити їм реальний доступ до візуалізацій, але обмеження полягає в тому, щоб змінити обладнання, щоб вони отримали доступ і могли бачити візуалізацію, але з цього дослідження ми проаналізували це, а не змінюємо апаратне забезпечення, якщо ми змінюємо наше програмне забезпечення. для створення різного роду візуалізацій для звичайних користувачів може бути корисно зрозуміти це та зробити його адаптивним відповідно до їхніх конкретних потреб.

Пізніше ми вивчали систему підтримки прийняття рішень та її можливості, що дає нам короткий опис архітектури та їх розвиток у реальному часі в різних областях. Наша наступна мета — створити адаптивний механізм, який фіксує поведінку користувачів, їхні симпатії щодо попередніх виборів і надає результати відповідно до їхніх вимог. У системі підтримки прийняття рішень такий тип адаптивної структури відсутній, пізніше буде введена ця структура візуалізації, яка забезпечує адаптовані результати візуалізації даних у системі підтримки прийняття рішень.

Мета нашої роботи полягає в тому, щоб продемонструвати, що адаптивний веб-підхід і весь його набір допоміжних механізмів, наданих WEBML, ECA правилом та іншими фреймворками, можна інтегрувати в платформу візуалізації

даних шляхом підвищення можливостей систем підтримки прийняття рішень, в яких візуалізація є важливою частиною DSS.

У цьому дипломному дослідженні нашою практичною метою є побудова адаптивної системи візуалізації даних, яка виявилася б інструментом підвищення продуктивності в системі підтримки прийняття рішень, де візуалізація є важливою частиною рішення, а також слугує основою для майбутніх досліджень. Ми зосередилися на вирішенні обмежень у попередніх структурах адаптації, а саме на попередньо визначених шаблонах адаптації, на умовах і якщо та зміні обладнання. Інша мета — створити структуру візуалізації даних, яка має можливості відображати складні форми двійкових наборів даних з можливістю розгортання на різних платформах і надає прямі докази з розширеними параметрами вибору користувача та багатобарвною візуалізації користувачам набору даних.

1.3 Аналіз відомих джерел предметної області

У сфері адаптації в електронному навчанні

Пов’язану роботу можна побачити в системі WEBML [3], в якій створена та працює платформа, яка є адаптивною за своєю природою і не залежить від будь-якої мови. Це відкритий фреймворк і може використовуватися з будь-якою системою. Вперше це було реалізовано в системах електронного навчання, які значною мірою стають адаптивними відповідно до вибору користувачів. Користувач спочатку входить в систему електронного навчання, щоб побачити різноманітні пропоновані курси. Якщо користувача не цікавить якийсь або деякі курси, система фіксує цю поведінку користувача. Це робиться виключно під час навігації від однієї сторінки курсу до іншої**.**

Коли користувач знову входить в систему, пропонуються лише ті курси, до яких користувач проявляє інтерес. Це реалізовано в електронному навчанні, оскільки це базовий характер адаптації.

Розвиток здійснюється на основі цієї архітектури. Повністю реалізовано обмеження зв'язків і часу, а також введено механізм правил, у якому фіксуються всі правила, і, крім того, програма працює за цими правилами.

У сфері адаптації в системі агентів

Інша пов'язана робота – JADE (Java Agent Development Framework) [5]. Запропонована платформа агентів використовує контейнери агентів, щоб агенти могли обробляти операції самостворення, самокерування та самознищення на основі контекстно-залежних служб з відфільтрованою та складеною контекстною інформацією монітора контексту [5]. У ньому також використовується менеджер реконфігурації, який динамічно переналаштовує платформу, шукаючи агентів і сервісів за допомогою операції виявлення.

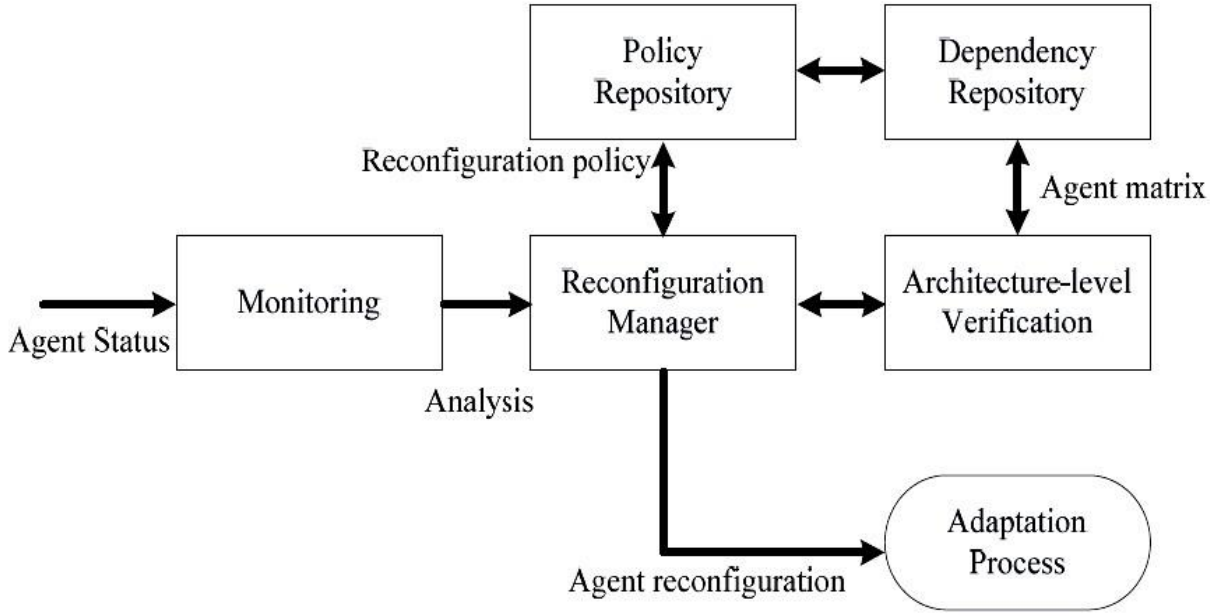


Рисунок 1.5 - Діаграми архітектури JADE

Підхід до адаптації під час виконання використовує переваги моделювання архітектури та кількох агентів. Він використовує модель аналізу залежностей на основі матриці, щоб гарантувати необхідну продуктивність після реконфігурації, яка була прийнята багатоагентною системою для автоматизації процесу

адаптації [5]. Менеджер адаптації агента складається з компонента аналізу, реконфігурації та перевірки на рівні архітектури для обробки реконфігурації агента під час виконання. Сам агент використовує матрицю fd для перевірки передумов продуктивності під час виконання. Менеджер реконфігурації є одним з абстрактних компонентів менеджера платформи. Компонент моніторингу використовує потік агента в контейнері агента для спостереження за станом агента. Він аналізує час виконання на всьому протязі з урахуванням обмежень продуктивності. Репозиторій політик керує стратегією адаптації за допомогою матриці залежностей. Менеджер реконфігурації перевіряє інформацію, витягнуту з даних моніторингу, а потім вирішує відповідну стратегію адаптації. Для цього існує попередньо визначена політика адаптації та прийняття рішень. Після цього він починає процес реконфігурації шляхом перетворення матриці. Репозиторій залежностей зберігає матричне представлення, що використовується у віддалених агентах, яке обробляється репозиторієм агентів для надання детальної інформації про реалізацію [5]. Після визначення політики перевірка на рівні архітектури перевіряє дійсність реконфігурації.

Таблиця 1.1 - Порівняння різних моделей

	НЕ адаптивний	Правило ECA	Рамка WEBML	Запропоновано
Адаптивність поведінки	Візуалізація Frame Work	Модель	Робота	Адаптивність
Залежність		Фіксовані правила	Фіксовані правила	Рамка візуалізації
Адаптаційний вік	Не знайдено	Зміни	Зміни	Робота
Відображення інформації про наведення миші	Не знайдено	Вручну	Вручну	Поведінка користувача

У таблиці 1.1 ми порівнюємо пов’язані роботи з поточною запропонованою структурою, ми ще раз заявляємо, що наша мета полягає в тому, щоб створити адаптовану структуру візуалізації, яка оптимізована для створення візуалізацій

даних із наборів даних, з розширеною візуалізацією на основі руху, інструментом прийняття рішень з мультиплатформою розгортання.

Функції, наведені в колонці пропонованої візуалізації адаптації, можна розглядати як список побажань, який необхідно реалізувати в нашій роботі.

1.4 Візуалізація даних

Візуалізація даних означає надання зображення існуючого програмного забезпечення за допомогою візуальних об’єктів. Візуалізація даних може візуалізувати різні аспекти даних, наприклад структуру даних, порівняння, потік і навіть певну поведінку тенденцій даних. Доведено, що відповідна візуалізація може значно скоротити зусилля, витрачені на різні етапи порівняння даних та розробку різноманітних рішень на основі типів даних. За допомогою візуалізації аналітики даних та зацікавлені сторони можуть отримати загальну точку зору на структуру даних, логіку даних або пояснити та спілкуватися з процесом підтримки прийняття рішень [10]. Як правило, візуалізація даних використовується для демонстрації поведінки даних, щоб можна було переглядати складні дані та порівнювати їх із простою візуалізацією. Численні дані доступні для вивчення та порівняння цих типів даних, щоб прийняти певні рішення за менший час. Візуалізація даних вважається найкращою технікою логічно і практично. Забезпечуючи гарне графічне представлення для візуалізації даних, краще розуміння системи користувачем може бути більш перспективним, ніж текстове представлення даних [9].

Завдяки графічному представленню даних можливості обробки даних користувача будуть використані набагато ефективніше, ніж інші методи представлення. Якщо відповідний інструмент графічного представлення вибрано правильно, буде менше потреби у записі перцептивної та розумової інформації, і це зменшить навантаження на пам’ять. Забезпечення графічних інтерфейсів забезпечить швидшу передачу інформації між комп’ютером і

людьми, оскільки було доведено, що користувачі можуть розпізнавати та класифікувати символи швидше та точніше, ніж текст.

Крім того, завдяки своїй простоті, графіка буде набагато легше залишатися в свідомості звичайних користувачів. Це також дає користувачам краще відчуття контролю, коли вони бачать об’єкти на екрані. Загалом, графічна візуалізація може надати багато інших переваг, таких як швидке навчання, швидше використання та вирішення проблем тощо.

Існує кілька доступних інструментів і методів візуалізації даних. Методи візуалізації складаються з набору елементів, таких як точки, лінії, форми, тексти та текстури, які кожен із цих елементів ілюструє як сутність або атрибут із набору даних, який буде візуалізувати. У деяких випадках для системи можна застосувати більше однієї техніки візуалізації. Методи візуалізації даних можна розділити на дві групи: статична візуалізація та динамічна візуалізація даних. Прикладом статичної візуалізації даних є візуалізація, утворена певними наборами даних у статичних формах, які формуються так само, як діаграми, а також представляють статичне значення. З іншого боку, динамічний — це той, у якому значення діаграми змінюється з певними часовими проміжками. Тут ми визначаємо часові рамки, які можуть бути часовими проміжками певних років або представляти один і той же час.

Існує різниця між значеннями двох стовпців, які згодом можна подолати або залишити незмінними.

Що стосується розмірності, то візуалізація може мати два або три виміри. Інструменти двовимірної візуалізації в основному включають графічні або деревоподібні уявлення, які можуть містити багато вузлів і дуг [11]. Для деяких систем, які мають занадто багато інформації для візуалізації, використання двовимірної техніки може призвести до плутанини. Тому в деяких роботах припускається необхідність додаткового просторового виміру, що може зробити більш можливим для дизайнера описати більше аспектів системи [11].

Щоб вибрати найкращу техніку візуалізації для наявних даних, перш за все, повинна бути зрозуміла причина і мета візуалізації. Потім слід визначити групу користувачів, рівень їх знань і досвіду роботи з комп'ютерними системами. Крім того, слід виявити всі наявні набори даних і елементи в наборах даних і всі зв'язки між ними, а також визначити аспекти даних, які будуть представлені. Необхідно також дослідити використання та обмеження існуючої системи, яку планується візуалізувати [12]. З іншого боку, необхідно оцінити існуючі методи візуалізації даних. Зрештою, слід вибрати та впровадити техніку, яка відображає найкращу візуалізацію наборів даних і відповідає вимогам системи.

Аналіз методів візуалізації даних

Основна причина такої візуалізації полягає в тому, щоб зробити текстову систему зрозумілою для користувачів. У цьому випадку користувачі витрачатимуть менше часу, щоб мати більш чітку та точну точку зору на систему. Таким чином, замість читання даних та запам'ятовування структури наборів даних, користувачі бачитимуть потік даних динамічно під час роботи з текстовою системою та матимуть огляд усіх наборів даних у великому масштабі в перед ними.

Особи, які приймають рішення, завжди використовують дані для визначення можливостей, тенденцій та проблемних сфер свого бізнесу. Більшість даних надходять до них у формі табличних звітів, і їм важко швидко й ефективно засвоювати інформацію, виявляти закономірності, виявляти відхилення та бачити приховані зв'язки. На щастя, хоча обсяг даних, з якими вони мають справу, постійно збільшується, інструменти візуалізації даних розвивалися до такої міри, що тепер вони можуть перетворювати велику кількість складних даних у значущі візуальні уявлення, які включають науку, що лежить в основі людського сприйняття та пізнання.

Основні діаграми є найпоширенішою та основною формою візуалізації даних. Основні діаграми включають кругові, гістограми та лінійні діаграми, які допомагають аудиторії швидко визначити загальні тенденції зростання та

зниження, а також групи даних, які працюють інакше, ніж інші групи. Ці базові діаграми зазвичай використовуються для відображення відносної ефективності ключових бізнес-вимірів, таких як сегменти ринку, групи продуктів, географічні регіони та бізнес-одиниці. Замість того, щоб дивитися на табличний список чисел, аудиторія може буквально побачити, як дані пов'язані з іншими даними. Ефективно висвітлюючи винятки, ці діаграми дозволяють аудиторії витрачати більше часу на пошук можливостей або вирішення проблеми і менше часу на пошуки чи визначення. Більшість онлайн-інструментів запитів і звітів створюють ці діаграми автоматично, тим самим скорочуючи час, який кінцеві користувачі витрачають на створення діаграм вручну в автономних інструментах, таких як електронні таблиці [13].

Графічні індикатори включають графічні значки, кольорові символи, стоп-сигнали, датчики та інші вимірювачі продуктивності, які використовуються для надання аудиторії швидкого візуального огляду, який переміщує фокус на інформацію, що представляє найбільший інтерес, і скорочує час, витрачений на вивчення цілого екрана інформації. [13]. Це було особливо корисно для покращення систем звітності за системою показників, які колись просто представляли швидкий перегляд табличного списку ключових показників.

Картки показників та інформаційні панелі все частіше включають основні діаграми для відображення підсумкової інформації та як спосіб представити додаткову інформацію, коли користувач вибирає певний показник. Наприклад, зіткнувшись із підсумковим показником, який вказує на те, що доходи є незмінними з року в рік, допоміжна гістограма може швидко повідомити, які регіони зростають, зменшуються чи не змінюються.

Розширені програми візуалізації даних забезпечують складні графічні представлення. Тоді як стовпчаста або кругова діаграма представляє просту серію точок даних. Розширені програми візуалізації можуть відображати більше півдюжини вимірів на об'єкт графіка. Ці графічні представлення виходять за межі традиційних кругових, стовпчастих і лінійних діаграм, включаючи

багатовимірні діаграми, гістограми, діаграми розсіювання, графіки сузір'їв, географічні карти, плани поверхів, просторові зображення, діаграми статистичного контролю та теплові карти серед інших графічних зображень. Оскільки компанії більше не обтяжені такими інструментами, як редактори базових звітів запитів і зведені таблиці (які можуть бути складними та неефективними у використанні, коли задіяно багато атрибутів даних), вони відкривають цінну інформацію про клієнтів, продукти, ринки, торгових представників, співробітників. , а також фінансові результати [13].

Анімація додає до візуалізацій ще один легко споживаний аспект. Спостереження за зміною даних з часом дає як швидкість, так і напрямок для значень, які вони представляють. Добре знати свої поточні продажі та витрати. Знання того, зростають вони чи зменшуються і якою швидкістю може бути критичним для бізнесу.

Занурюючі рішення для візуалізації даних поміщають аудиторію у віртуальне середовище, що містить тривимірні представлення складних даних.

Середовища варіюються від систем розміру столу до розміру стіни до кімнат розміру та поєднують технології візуалізації, обчислень, керування даними та проекційних технологій. Візуалізація даних із зануренням широко використовується в сферах досліджень та розробки продуктів, де користувачі отримують вигоду від практичної інтерактивності з віртуальними моделями, прототипами та таблицями даних. Уявіть, що ви перебираєте багатовимірні набори даних про продажі або витрати, щоб визначити тенденції, закономірності та відхилення [13].

Інтерактивні можливості Взаємодія є ключем до надання повної інформації. У той час як традиційні середовища звітності представляють собою статичні кругові та стовпчасті діаграми, інтерактивні програми візуалізації дозволяють користувачеві безпосередньо взаємодіяти з діаграмою або зображенням, щоб вивчати основні дані та відкривати нову інформацію. Діаграми або зображення можна обертати, щоб отримати найбільш глибокий огляд інформації. Частини

зображення можна переміщати мишею, щоб відкрити контекстну інформацію про точку даних, включаючи значення даних і різні атрибути розмірів, такі як географічне розташування, демографічна група, організаційна сутність тощо. Смуги прокручування можна використовувати для переміщення вліво, праворуч, вгору та вниз для перегляду точок даних, які в іншому випадку були б поза екраном через обмеження традиційних інструментів стовпчастих і лінійних діаграм.

Площини фільтрів дозволяють користувачеві ізолювати точки даних вище порогового значення, подібно до гірських вершин, що піднімаються над низькорівневими хмарами [13]. Це привертає увагу аудиторії до цінностей, що викликають найбільший інтерес. Ці можливості аналогічні візуальному запиту, де користувачі можуть вибирати підмножини даних, збільшувати їх до деталей, фільтрувати небажані значення та детально вивчати області, що представляють інтерес, так само, як і з запрограмованим запитом, але без необхідності приймати рішення заздалегідь. на яких розмірах даних зосередитися. Ця безпосередня взаємодія з візуалізацією дозволяє користувачеві перемістити фокус з написання запитів, форматування звіту та інтерфейсу користувача на виявлення та аналіз інформації [13].

Зв’язок інформації між елементами дисплея за допомогою кольорового кодування та синхронізованого контексту є однією з найпотужніших передових інтерактивних можливостей. Це дає багатший порівняльний контекст, що полегшує використання інформації та взаємодію між зображеннями, оскільки користувач робить різні аналітичні рішення.

У цьому розділі ми вивчаємо та аналізуємо різні методи візуалізації даних. Оскільки візуалізація даних — це величезна область, тому перед початком роботи з фреймворком необхідно розібратися в теорії візуалізації. Як насправді візуалізація допомагає реальним користувачам продемонструвати різні факти сукупних даних. У цьому розділі аналізуються різні аспекти, пов’язані з

вивченням візуалізації, шляхом надання інформації та можливостей, що пропонуються методами візуалізації даних.

Ці методи надалі реалізовані в нашій роботі над реальною структурою візуалізації. З цих методів ми впроваджуємо в наш фреймворк, щоб ця адаптивна структура фактично працювала і надавала результати користувачам у реальному часі. Далі ці візуалізації аналізуються і додаються лише ті візуалізації, які фактично інтерпретують і фактичне значення наборів даних і які зрозумілі людині, оскільки ці візуалізації можуть відігравати важливу роль у системі підтримки прийняття рішень.

2 ПРОПОНОВАНА АДАПТИВНА ОСНОВА ВІЗУАЛІЗАЦІЇ ДАНИХ

2.1 Архітектура системи адаптивної візуалізації даних

Раніше розроблені фреймворки візуалізації здатні створювати різноманітні візуалізації даних. Ці фреймворки призначені для створення певних типів діаграм і візуалізації, які можуть відображати набори даних у графічні форми. Вони керують наборами даних і можуть візуалізувати їх відповідно до графіків, запитаних або розроблених користувачами, щоб показати кінцеві результати. Зі зміною наборів даних змінюється і форма візуалізації. Користувачі можуть переглядати лише графіки, призначені для відображення результатів. Пізніше користувачі можуть переключатися з одного типу візуалізацій на інший тип.

Покращенням, яке можна застосувати до структури візуалізації даних, є поведінка системи. У цій дипломній роботі представлена адаптивна візуалізація даних, яка візуалізує набори даних і дає кінцеві результати, які найкраще обслуговуються в своєму класі. Він використовується в системах підтримки прийняття рішень, де для перегляду найкращих результатів заданих наборів даних потрібні критичні за часом події, які найкраще обслуговуються та вибираються користувачами та адаптивно створюються. У попередньому розділі вже згадувалося, що методи візуалізації вважаються найкращим способом отримання результатів великої кількості наборів даних. Крім того, цей фреймворк є незалежним компонентом, який можна застосовувати і використовувати в багатьох системах і не вимагає специфічних мовних вимог.

З цієї причини цей метод візуалізації показує адаптивність і можливість розгортання на різних платформах. Візуалізація виконується на сервісному шарі та зверху на них. Користувачі задають різні запитання, щоб зібрати інформацію про користувача, а також зібрати журнал часу, проведеного користувачами на екранах. Пізніше ці журнали використовуються для виробництва адаптивних речовин в системі. Існують різні підходи та фактори, які також сприяють адаптивності системи. Зараз для візуалізації використовуються двовимірні

діаграми, але для візуалізації наборів даних це можуть бути нескінченні набори даних, які можна візуалізувати на фреймворку, а не виконувати запити до бази даних. Також усі діаграми є графіками руху, які змінюються з часом. Таким чином, ці діаграми також можна використовувати як один інший вимір, тобто час. Мета цієї адаптивної системи візуалізації даних полягає в тому, щоб дати користувачеві, який переважно приймає рішення, краще розуміння наборів даних і допомагає швидко приймати рішення там, де важливі за часом події є основною частиною системи.

Обговорення, розглянуті в попередньому розділі, дають змогу побачити архітектуру системи з наступними характеристиками:

• Платформа візуалізації даних на основі графіків для представлення баз даних. Ця платформа підходить для представлення кількох наборів даних у графічній формі для візуалізації даних від наборів чисел до певних цифр. Ця платформа не залежить від бекенд-технологій і може бути реалізована в будь-якій системі.

• Представлено механізм адаптації користувача, який використовується для інтелектуальної візуалізації наборів даних, що найкраще підходить у своїй категорії. Ця система адаптації здатна адаптувати середовище та автоматично змінюється, щоб надати бажану користувачеві візуалізацію для певного типу наборів даних.

• Існує підхід культурного лідера, який дозволяє системі працювати незалежно та розумно від іншої підкатегорії візуалізації, а також це впливає на адаптацію відповідно до їхньої культури та поведінки.

• Це визначає два алгоритми – один призначений для адаптації, а інший – для представлення різної візуалізації одних і тих самих наборів даних двом різним користувачам.

• Побудова та керування ефективною моделлю користувача, яка дозволяє системі надати користувачеві найбільш відповідне візуальне уявлення відповідно до його чи її навичок та потреб.

2.2 Архітектура системи

На рисунку 2.1 показано архітектуру системи, яка включає платформу візуалізації.

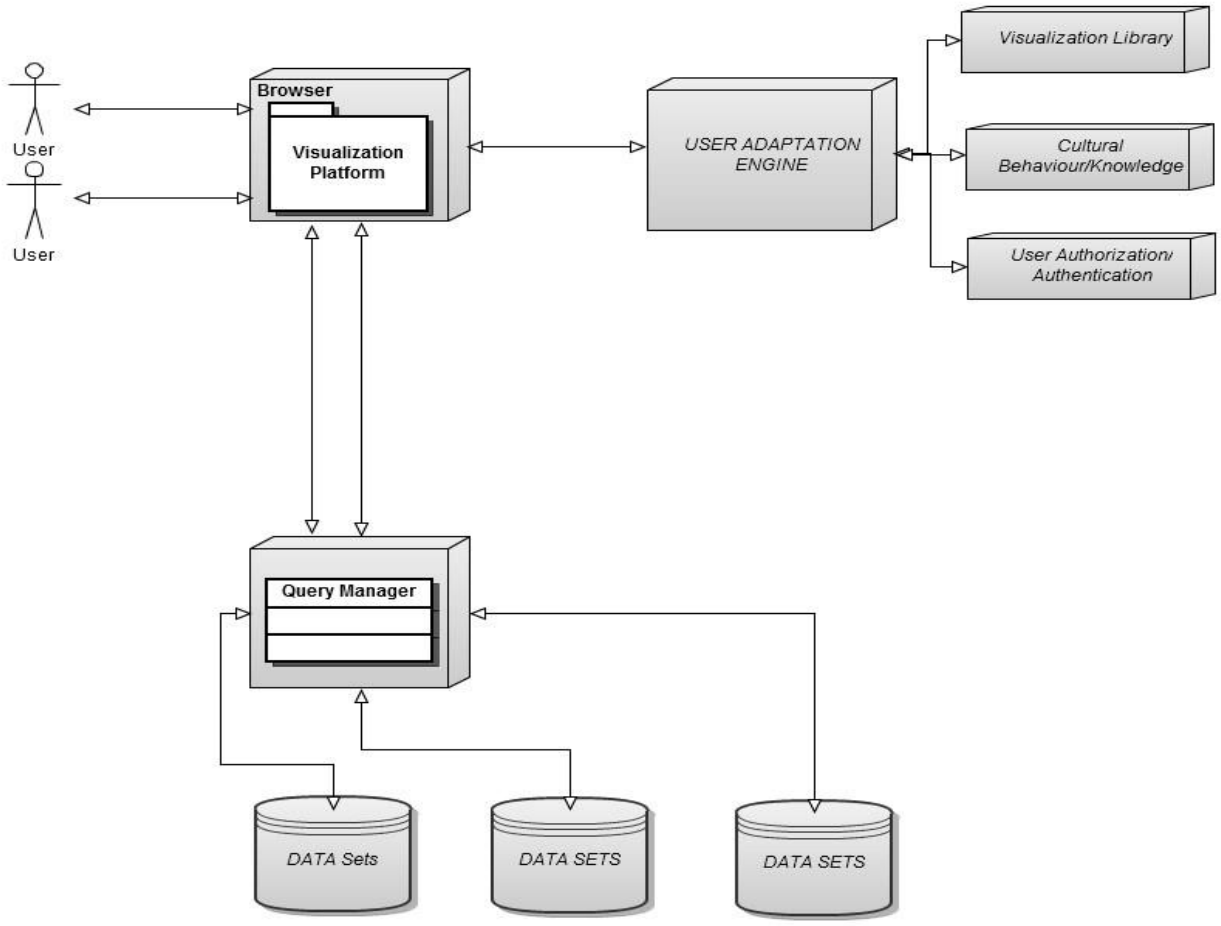


Рисунок 2.1 - Архітектурна схема фреймворку

Механізм адаптації користувача, який включає бібліотеку візуалізації, культурну поведінку/знання, авторизацію/автентифікацію користувача. Також є менеджер запитів, який підключається до наборів даних для отримання необхідної інформації з бази даних шляхом виконання певних запитів. Механізм адаптації користувача відповідає за адаптацію шляху налаштувань користувача, оскільки користувач взаємодіє з системою, оновлює адаптивний шлях і надає найкращий шлях іншим користувачам відповідно до їхніх запитів. Адаптація

користувача достатньо розумна, щоб забезпечити найкращу візуалізацію користувачам, адаптовану користувачами. Система адаптації користувача використовує штучний інтелект, відображаючи лише візуалізації відповідно до якості даних, якість даних означає, що існує велика кількість доступних наборів даних, але всі вони не можуть використовуватися в одних і тих же візуалізаціях, тому для цього механізму адаптації користувача перевіряється які візуалізації найкраще обслуговуються для заданої якості даних. Крім того, механізм адаптації користувача відповідає за рекомендацію користувачам візуалізації, вибрану експертними користувачами, або ми можемо сказати, що експерти-користувачі впливають на людей, які можуть змінювати шаблон адаптації та впливати на інших користувачів, використовуючи свій інтелект на задані запити.

Роль користувача

Адаптивний фреймворк візуалізації даних призначений для роботи в різних організаціях різної сфери роботи. Кожна організація має політику цілісності даних для різних рівнів користувачів, оскільки модуль цієї ролі користувача розроблений у рамках. Кожен рівень користувачів має доступ до візуалізації відповідно до потреб певної ролі. Тільки користувачі зі спеціальним дозволом на візуалізації можуть отримати доступ до всіх видів бібліотек візуалізації. Таким чином, він працює виключно на основі рольової моделі, яка працює як приклад для наборів діаграм:

Уся діаграма= {A}, лінійна діаграма= {L}, стовпчаста діаграма= {C}, бульбашка = {B}

Користувач X = {C} {B}, користувач Y= {A} ∩ {C}

Вводячи роль користувача в нашу структуру, ми можемо легко керувати профілем користувача, і кожен користувач може мати спеціальні дозволи на візуалізацію, оскільки для певної ролі користувача їм потрібен лише один тип діаграми.

Крім того, у нашій системі візуалізації даних кожен користувач має різну роль, і механізм адаптації користувача достатньо розумний, щоб фіксувати

особисту поведінку користувача на тих самих даних або може сказати, щоб запам’ятати вибір користувача для того самого типу даних, тому, коли запит виконується до бази даних, перевірка механізму адаптації користувача про вибір візуалізацій, які використовувалися користувачами в попередніх виборах для того самого типу наборів даних, і надати користувачам ці візуалізації.

Маркування даних

Позначення даних визначається як маркування наборів даних, так що, коли набори даних переглядаються кінцевими користувачами, вони надають інформацію про набори даних у кінцевому значенні. У візуалізації дані представлені у вигляді стовпчастих, кругових діаграм, бульбашкових діаграм тощо. Коли кінцевий користувач переглядає результат кінцевого запиту, дані представлені у вигляді діаграм, і кожен користувач інтерпретує дані по-різному. Тому маркування даних є обов’язковим у формі візуалізації. У нашому фреймворку, коли користувач наводить курсор миші на візуалізацію, значення даних представляються з максимальною інформацією, яка може бути доставлена кінцевому користувачеві. Таким чином, маркування даних надає всю відповідну інформацію разом із іншою інформацією.

Запит

У фреймворку конструктор запитів працює з фреймворком. Виконуючи один запит, користувач може взаємодіяти з кількома візуалізаціями. Виконуючи один запит, який виводить у вигляді значень даних, користувач може побачити візуалізацію окремого набору даних кількома способами. Користувачеві не потрібно щоразу запускати запити, щоб побачити різні візуалізації.

Типи діаграм

У фреймворку доступні різні типи діаграм. Для відображення даних у кількох формах потрібні різні типи діаграм. Відповідно до вимог користувача, типи діаграм та аутентифікація надаються кінцевому користувачеві. Ці типи діаграм можна пізніше додати в систему. Різноманітні доступні типи діаграм, які можуть бути реалізовані в системі, – це діаграми площ, лінійні

Діаграми, стовпчасті діаграми, стовпчасті діаграми, кругові діаграми, бульбашкові діаграми, свічка, кілька серій, графіки, каскадні стовпці, кілька даних і діаграми HLOC.

2.3 Багатомовне розгортання

Цю структуру візуалізації можна розгорнути на багатомовних фреймворках, оскільки ця система може використовуватися в багатьох багатомовних комп'ютерних системах. Запропоновану систему можна ввімкнути для роботи з java, C++, vb.net тощо, оскільки цей фреймворк призначений для роботи на різних платформах для надання візуалізації без зміни мови. Будь-яка мова може надіслати запит на відображення візуалізація.

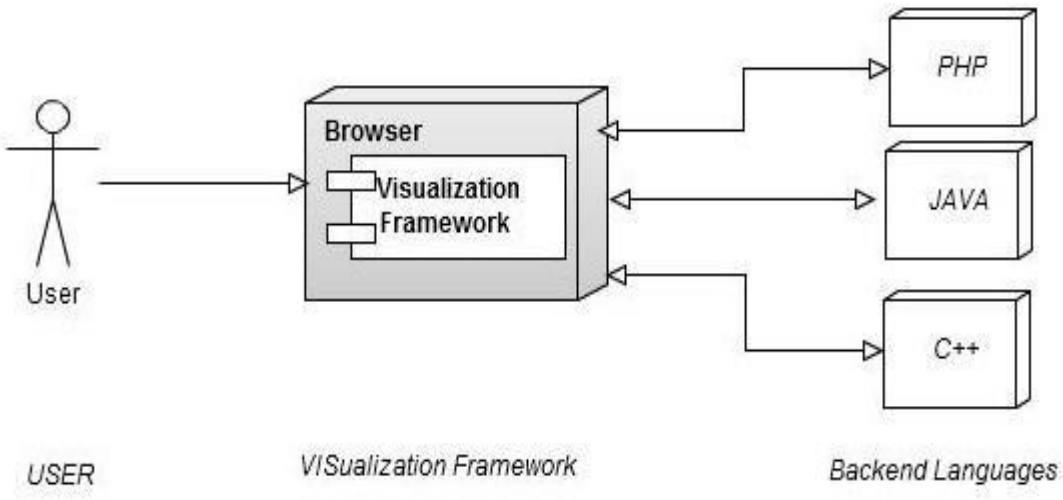


Рисунок 2.2 - Багатомовна схема розгортання системи

Механізм адаптації користувачів відповідає за надання адаптованих результатів за допомогою підходу штучного інтелекту. Система адаптації користувачів фіксує поведінку користувача щодо вибору, вибраного користувачами в наборах даних. Він фіксує та відображає ті самі варіанти візуалізації, коли вибрано однаковий тип наборів даних. Крім того, він розумно рекомендує іншому користувачеві щодо вибору, обраного досвідченим

користувачем. Експертні користувачі – це користувачі, які впливають на інших користувачів; результати, вибрані цими експертними користувачами, мають сильний вплив на створення деяких впливових результатів. Також перевіряється якість даних, і система оптимізується для створення лише тих візуалізацій, які найкраще обслуговуються для даних.

```
For Each Query Q is done
Display Results in table T
IF V is essential part of the query
 Call displaychart with V
   t = display V
 Check user Authorization
   t=check user group
 Show All visualizations
 else show authorized visualizations
Check Query name
     t= compare query name in database
IF exists check adaptivenes factor A
Display result with max A
ELSE Check Table name elements
     t= check table name elements in query table
 IF exists check adaptivenes factor A
Display result with max A
     t=display result with maximum adaptivenes factor
ELSE  Check user recent charts
Display result with max viewed
Else Display default
LOG user data
     t=createlogs of user session
with time t max(t) and min (t)
with selection of chart for given query Q
Update Adaptivenes factor A for given query Q
```

Рисунок 2.3 - Алгоритм роботи фреймворка

Нижче наведено пояснення вищезазначених алгоритмічних кроків. У системі можуть статися такі випадки.

a. Для кожного запиту Q виконується в системі бази даних з певними атрибутами

б. Результати відображення запиту в таблицях T із рядком і стовпцями.

c. Якщо є необхідність візуалізувати результати запиту там, де це є важливою частиною запиту, викличте діаграму відображення функції. Завдання – відобразити результат у формі візуалізації.

d. Тепер перевірте авторизацію, чи має користувач право переглядати всі діаграми або певні конкретні діаграми. Функція виклику для перевірки групи користувачів.

e. Тепер перевірте Ім’я запиту, якщо таке ім’я з’являється в системі бази даних

f. Якщо є така ж назва, перевірте адаптивний фактор

g. Для коефіцієнта максимальної адаптивності відобразіть тип діаграми, який має максимальні адаптивні властивості

ч. Якщо немає коефіцієнта адаптивності, логіка програми перевіряє останні діаграми, які переглядали користувачі, а її коефіцієнт інтелекту перевіряє, які діаграми подобаються користувачеві.

я Якщо попередньої історії немає, то відображається діаграма за замовчуванням.

j. Тепер ведення журналу виконано, яке фіксує час користувачів, вибір та адаптивність оновлення фактор.

У цьому розділі представлена запропонована нами система. Наша перша мета — створити архітектуру системи, як ми раніше вивчали різні адаптивні фреймворки та візуалізацію наборів даних. Тепер наша головна мета — створити каркас, насамперед, для системи підтримки прийняття рішень, яка здатна надавати візуалізацію даних за допомогою адаптивних технологій. Введена архітектура запропонованого фреймворка, яка має можливість створювати різні

типи діаграм для різних ролей у системі з розширеними параметрами вбудованого модуля запитів і маркування даних, яка також працює зі штучним інтелектом і може бути розгорнута на кількох платформах для отримання адаптивних результатів візуалізації даних. . Пізніше алгоритм для цього введено в главі, яка покроково пояснює деталі роботи. У цій главі пояснюються технічні деталі та дизайн архітектури на основі цієї фактичної реалізації, що пояснюється в наступній главі разом із деталями тестування зручності використання.

2.4 Реалізація

У цій дипломній роботі для розробки «редактор Eclipse» використовувався для проектування та розробки компонента діаграми. Усі компоненти діаграми використовують Flex engine, програмне забезпечення з відкритим вихідним кодом. Використаним сценарієм був Macromedia Flex Markup Language, також відомий як MXML, і Action Script 3 для інтерфейсу користувача та логіки діаграм. Усі діаграми візуалізації даних, включаючи лінію, стовпець, круг, бульбашку та діаграму, були спеціально розроблені з використанням Action Script 3 та MXML, без використання стороннього API. Вихідний код проекту можна відкрити та налагодити за допомогою Adobe Flex Builder або Eclipse Editor.

В основному два типи вихідних файлів існують у flex-проекті (1) MXML (2) AS, а скомпільований файл – це флеш-файл SWF, який працює за допомогою плагіна Flash Player, встановленого у веб-переглядачі або на комп'ютері. Оскільки це фреймворк без мов, це означає, що його можна використовувати де завгодно з будь-якою серверною мовою. Таким чином, він має власну базу даних та компоненти адаптації, які є незалежними за своєю природою. Отже, щоб почати роботу цього компонента, потрібна структура даних xml, яка живить компоненти діаграми, після чого вона починає працювати самостійно.

Компонент діаграми починається із завантаження каналу XML за допомогою сценарію дії. Після завершення завантаження XML виконується синтаксичний аналіз XML для зберігання даних XML в об’єктах/колекціях сценарію дії, який служить постачальником даних для всіх діаграм. Ефекти анімації реалізовані під час завантаження даних у діаграми для плавного малювання діаграми. У правій частині області діаграми відображається список категорій з відповідним кольором категорії діаграми. На обох осях відображається набір параметрів для вибору різних осей/параметрів даних для діаграм. Коли відображення діаграми завершено, у користувача випадковим чином запитується введення певної візуалізації запиту, яке потім зберігається і реєструє дані для адаптації. Спочатку адаптація для всіх діаграм і системи дорівнює нулю, але коли користувач взаємодіє з системою, адаптація починає реєструвати дані, які зберігаються у файлах бази даних. Пізніше ці файли бази даних використовуються для цілей штучного інтелекту для відображення діаграм, які найкраще обслуговуються для заданих запитів.

Основний початковий файл: DevChartComp.mxml і він починається з тегу <application>. Це перший тег або файл, який виконується як початкова точка візуалізації. При ініціалізації програми,

Подія «ініціалізація» запускається, і ми викликаємо функцію сценарію дії («appInit()») для цієї події, щоб завантажити XML-канал. Коли ця подія ініціалізується, вона починає завантажувати всі відсортовані дані з файлів XML. Увімкнено

Завантаження XML, завершена програма викликає функцію «loadDataComplete». На цьому етапі програма виконується із завантаженням даних з подальшим аналізом XML для збереження даних в об’єктах сценарію дії за допомогою функції «mXMLToObject». Після того, як додаток завершено з компонентами дизайну інтерфейсу користувача, які відображають створення, повна подія запускається, і програма викликає іншу функцію «mInit». У цій функції виконується базовий стиль компонентів, і ця функція забезпечує

відтворення всіх компонентів інтерфейсу користувача та призначення їм даних, на цьому етапі завантаження більше не очікується.

У коді MXML ми використовували елемент керування Tab Navigator для створення перегляду вкладок різних діаграм

Перший компонент, тобто лінійні діаграми, відображається за замовчуванням. Кожна діаграма має властивість

«dataSet», який служить постачальником даних для цієї діаграми і прив’язаний до джерела даних, тобто «valueObj».

З обох боків осі діаграми ми використали елемент керування зі списком, який надає можливість перемикати категорію даних для цієї осі. Щоразу, коли користувач робить будь-який вибір у полі зі списком, ініціюється подія «зміна» цього поля зі списком, і ми викликаємо спеціальні функції «updateXAxis()» та «updateYAxis()» для відповідних елементів керування вікном зі списком. Ці відповідні функції використовуються для оновлення компонентів діаграми для нового набору даних. Кожен компонент діаграми має функцію «onCreationComplete()»; це використовується для створення відповідної спеціальної діаграми та призначення даних. У правій частині компонента діаграми ми маємо елемент керування «Легенда», який використовується для відображення списку категорій. Для керування швидкістю ефекту анімації діаграм у нас є горизонтальний повзунок, де користувач може встановити швидкість анімації від 1 до 10 секунд.

Усі діаграми — це спеціальні динамічні діаграми, створені за допомогою Action Script 3 на основі даних, що надходять із XML. Кожна діаграма має спеціальну панель підказок для розміщення додаткових інформація, що стосується будь-якої точки даних діаграми.

Усі ці діаграми є розширеними елементами керування базовими елементами керування діаграмами, наданими Adobe flex SDK, які налаштовуються за допомогою сценарію дії відповідно до вимог.

Тепер настає частина адаптації, в якій користувача просять випадковим чином надати перегляди та рейтинг для кожної діаграми, коли кожен з них відображається. Також реєструється час, витрачений на кожну діаграму. Таким чином ми отримуємо дані та зберігаємо дані у файлі db для кожної діаграми. Пізніше, коли буде виконано той самий тип запитів, наша програмна логіка перевіряє, яка діаграма має найбільші адаптивні значення. Якщо він його знайде, то ця діаграма відображається першою, але якщо ні, він логічно перевіряє різні фактори, такі як уподобання користувача в останній раз, який тип діаграм користувачеві подобається найбільше, а також інші елементи в таблицях діаграм, щоб забезпечити відображення діаграми. відповідно до якості даних.

2.5 Дослідження юзабіліті

Юзабіліті – це характеристика якості, яка оцінює деякі основні атрибути в процесі розробки програмного забезпечення. Вважається, що атрибути юзабіліті – це деякі точні й вимірювані компоненти абстрактного поняття, яким є юзабіліті [14]. Ці атрибути:

• Здатність до навчання – наскільки швидко і легко користувачі можуть виконувати продуктивну роботу з новою системою і як легко вони можуть запам'ятати, як система працює, після того, як деякий час не користувалися системою.

• Ефективність – кількість завдань, які може виконати користувач за певний час інтервал.

• Надійність – рівень помилок у системі та час, необхідний для відновлення після помилок.

• Satisfaction – рівень задоволеності користувача після роботи з системою.

Ці атрибути можна виміряти шляхом спостереження за користувачами, коли вони працюють із системою, та інтерв’ю з ними після того, як вони виконають завдання з системою [14].

У питаннях взаємодії людини з технологією парадигми аналітичного дослідження недостатньо. Тому емпіричні дослідження в галузі програмної інженерії постійно стають все більш прийнятними [15]. Юзабіліті – це те, як система взаємодіє з користувачем [16]. Інженерія юзабіліті визначає кінцевий рівень юзабіліті та гарантує, що програмне забезпечення, яке тестується, досягне цього рівня [16].

Дослідження юзабіліті можна проводити різними методами. Ці методи поділяються на дві загальні групи: емпіричні та аналітичні. У проектах, у яких людина взаємодіє з машинами, дуже корисні емпіричні дослідження [15]. З іншого боку, аналітичні дослідження можуть дати ранні відгуки про дизайн інтерактивної системи тестувальникам програмного забезпечення. Аналітичний метод складається з двох класів методів, якими є перевірка зручності використання та когнітивні покрокові інструкції. Перевірка зручності використання передбачає систематичну перевірку дизайну за допомогою деяких факторів для практичного, хорошого дизайну.

Одним із прикладів методу перевірки зручності використання є евристична оцінка, яка є неформальним методом тестування юзабіліті. У цьому методі, заснованому на рекомендаціях щодо проектування загального призначення, оцінювач перевіряє запропонований дизайн, щоб перевірити, чи були дотримані принципи зручності використання в дизайні [17].

У [17] запропоновано дев’ять евристик: простий і природний діалог, розмовляти мовою користувача, мінімізувати навантаження на пам’ять користувача, бути послідовним; забезпечити зворотній звязок; забезпечити чітко позначені виходи; забезпечити короткі шляхи; гарні повідомлення про помилки та запобігання помилкам.

Інший метод, який є когнітивними покроковими інструкціями, використовує більш чітку, детальну процедуру та проводить аналіз юзабіліті, заснований на роботі, тестуючи реальних користувачів, коли стикаються з системою. Він аналізує якість інтерфейсу, спрямовуючи користувача на

виконання завдання, задаючи три прості запитання: чи буде правильна дія достатньо очевидною для користувачів?, чи пов’яже користувачі опис правильної дії з тим, чого вони намагаються досягти?; Чи правильно інтерпретують користувачі відповідь системи на обрану дію?. Щоразу, коли на будь-яке з цих питань є відповідь «ні», можуть виникнути проблеми [17].

Існує інженерна модель юзабіліті, представлена «Гулдом і Льюїсом» і називається «відомі правила». Правила, що входять до нього: раннє зосередження на користувачах, участь користувачів у проектуванні, координація різних частин інтерфейсу, емпіричне тестування користувачами та ітераційний перегляд проектів на основі результатів тестування [18].

Для інженерії юзабіліті існує термін, який називається життєвим циклом інженерії юзабіліті, який означає не тільки те, наскільки задовольняє поточний дизайн інтерфейсу, але й чи можна його змінити для майбутнього інтерфейсу. Цей життєвий цикл складається з трьох етапів:

Попередньо розроблений етап

Основна мета цього кроку – дізнатися про цільового користувача та завдання, яке він збирається виконати. Чим більше це зроблено на цьому етапі; це було б більш рентабельним для всього тестування, тому що, швидше за все, кількість змін, які слід зробити в майбутньому, буде зменшено. Раннє зосередження на користувачеві. Перший крок у тестуванні юзабіліті — це дізнатися про користувача та його точні потреби від системи. Наприклад, знання про роботу користувача , досвід, рівень освіти, вік, рівень комп’ютерного досвіду допоможуть передбачити проблеми користувача і, відповідно, допоможуть створити кращий інтерфейс з урахуванням труднощів у навчанні користувача. Також слід виявити слабкі сторони нинішньої системи. Повинно бути зрозуміло, що в нинішній системі, які перешкоди виникають у користувача на шляху до досягнення цілей, що змушує користувача витрачати багато часу або викликає незручність у системі.

Встановлення цілей юзабіліті: під час тестування юзабіліті повинні бути виконані чотири характеристики юзабіліті. Очевидно, що для кожної системи пріоритет кожної характеристики буде різним. Але в цілому, оскільки всі вони пов’язані між собою, отримання хорошого результату в будь-якому з них може принести задоволення.

Етап проектування

Основна мета цього етапу — досягти придатної для використання реалізації, яка придатна для випуску. З цієї причини спочатку, на основі принципів зручності, ми повинні надати прототип остаточної системи, а потім протестувати прототип системи з реальними користувачами, щоб переконатися, що дизайн відповідає нашій меті.

1. Участь користувача в проектуванні.

2. Координація різних частин інтерфейсу

3. Емпіричне тестування користувачів. Цей крок дуже корисний і має дві основні форми. Перший, здебільшого кількісний, перевірить, чи досягнута ціль зручності використання чи ні, а другий, більш якісний, шукатиме причину неправильних частин інтерфейсу та кількість їх невірності. буде з'ясовано. На цьому етапі можна використовувати різні методи тестування, як-от спостереження користувача під час роботи з системою та запитання користувача про його досвід роботи з системою.

4. Ітеративний перегляд дизайну на основі результатів тестування: на основі етапу емпіричного тестування ми можемо перепроектувати інтерфейс і знову провести тестування на новому інтерфейсі.

Етап постпроектування

Цей етап є подальшим вивченням продукту, який використовується в цій галузі. Те саме завдання, яке виконується на стадії проектування, що полягало в перегляді дизайну та неодноразовому його перевірці, можна виконати з кінцевим продуктом, розглядаючи кінцевий продукт як прототип майбутньої продукції [19].

2.6 Пропонований метод тестування юзабіліті

На основі того, що обговорюється в розділі про зручність використання цього дослідження, користувачам було надано опитування. У цій дипломній роботі було проведено як аналітичне, так і емпіричне тестування. Для аналітичного тестування було використано комбінацію методів перевірки зручності та когнітивних покрокових інструкцій разом із відомими правилами, наскільки це було застосовно та практично з наявними можливостями, на різних етапах проектування системи. Основна увага була зосереджена на досягненні атрибутів зручності, наскільки це можливо, в рамках візуалізації.

Система візуалізації даних модифікована шляхом застосування відповідних запитів до наборів даних. Ці запити вже виконані для наборів даних і остаточні результати наборів даних, які потім візуалізуються в нашій структурі. Користувач у системі міг переглядати запит і результати. Потім усі результати візуалізуються, з яких користувач може вибрати різні доступні візуалізації. Коли користувачі переглядають результати, вони можуть дати зворотний зв’язок, з якого починається частина адаптації. Наступного разу користувачам буде показано результат на основі цього запиту, який раніше вибрано як найкращий. Для перевірки всі дані реєструються в системі, і ці дані потім використовуються для збору результатів у різних формах. Близько 10 запитів виконується до різних форм наборів даних. Ці запити спочатку відображають результати у вигляді лінійних діаграм за замовчуванням, а після адаптації системи ми можемо бачити різні діаграми для різних запитів як діаграми за замовчуванням. Дані в журналі потім використовуються для отримання результатів, оскільки через певний проміжок часу користувачам подобається той самий результат конкретного запиту. Коментарі користувачів реєструються, щоб отримати більше результатів. Це дослідження збирає факти про користувача, щоб прийняти певне рішення в системі підтримки прийняття рішень, просто переглянувши нашу систему візуалізації. Переглядаючи певні результати візуалізацій, які найкраще

обслуговуються для даного запиту, користувач може прийняти певні рішення, пов’язані з критично важливими подіями.

Для емпіричного тестування обидві системи були перевірені користувачами як з інформатики, так і з бізнес-відділів. Користувачам було надано анкету, щоб запитати їхню ідею про систему як для покращення зручності користувальницького інтерфейсу, так і для порівняння системи візуалізації з простою системою результатів даних. Крім того, у деяких особливих випадках ідею деяких користувачів з великим досвідом розробки програмного забезпечення запитували та застосовували до системи, наскільки це було можливо.

3 ВЛАСНЕ ОРИГІНАЛЬНЕ АЛГОРИТМІЧНЕ ТА ПРОГРАМНЕ РІШЕННЯ ЗАДАЧІ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

3.1 Складові програмного рішення

Наша система візуалізації даних була піддана оцінці зручності використання двома групами користувачів з різним рівнем аналізу даних та комп’ютерної експертизи. До першої групи увійшли 20 студентів з дуже невеликим досвідом використання систем підтримки прийняття рішень, але хорошими навичками оцінки даних бізнес-відділу. Другу групу склали 50 студентів інформатики. Передбачалося, що загалом студенти з інформатики матимуть більший досвід використання програмного забезпечення/інструментів візуалізації даних, ніж студенти бізнес-класу, від яких очікується, що вони не матимуть жодного або дуже малого досвіду в цій галузі.

Таблиця 3.1 – Розподіл учасників тесту за їхнім академічним рівнем

	комп'ютер Студенти	Бізнес Студенти
	Візуалізована система	Візуалізована система
Бакалавр (1 курс)	3	1
Бакалавр (2 курс)	9	2
Бакалавр (3 курс)	4	6
Бакалавр (4-й курс)	4	10
магістра	24	1
PhD	6	0

3.2. Опис фреймворку для користувачів

Порядок виконання завдань, які учасникам було запропоновано виконати, такий. Спочатку їм було запропоновано пройти короткий опис адаптивної системи візуалізації даних. Учасник може зрозуміти структуру візуалізації. Їм було надано короткий введення в адаптивну частину щодо того, як система стає адаптивною, коли користувачі користуються системою. Також їм було надано детальну інформацію про те, як опитування розроблено для вимірювання історії, досвіду та задоволеності користувачів у системі підтримки прийняття рішень. Було близько 10 запитів різних наборів даних, які були майже виконані на різних унікальних базах даних і в основному були з наборів даних канадської статистики [20]. Ці результати запиту потім були візуалізовані за допомогою нашого інструменту адаптивної візуалізації даних. Адаптивний інструмент візуалізації даних візуалізує всі результати запитів у різних типах діаграм. Користувач міг досліджувати різні типи діаграм і вибирати найкращий результат, ставити «подобається» і «не подобається». Також реєструвався час, витрачений на кожну діаграму. У кожному запиті було надано коротке введення про запит, а також для реєстрації даних, задавалися запитання, щоб користувач міг прочитати набори даних і міг думати та відповідати на ті самі запитання. Надані відповіді також реєструвалися. Опитування проводилося після одного та іншого запитання, і всі дані були записані та збережені. На початковому рівні опитування всі діаграми були налаштовані на прості лінійні діаграми. Оскільки користувачі використовували систему та давали свої відгуки, система отримувала адаптивний та відображався результат, обраний користувачами та який найкраще показав свій клас. Оскільки системи підтримки прийняття рішень можуть використовуватися в різних сферах і в різних сферах знань, ці системи мають різні результати в різних дослідженнях та опитуваннях.

3.3. Завдання етапу проектування

Після опису фреймворку користувачі почали своє опитування. Їм надали інструмент опитування, який складався з різних запитів у різних галузях. Користувачі почали використовувати опитування та відповідали на один запит за іншим. На одній сторінці був лише один запит. Користувачеві, який тестував інструмент візуалізації даних, вже було надано деталі всіх наборів даних. У деяких запитах користувачів просили вибрати, яка візуалізація забезпечує найкращий перегляд наборів даних відповідно до порівняння двох даних в одній візуалізації. У деяких випадках користувачі висловлювали свою думку щодо тривожних факторів для всіх наборів запитів. Як і при розгляді систем підтримки прийняття рішень, запити обслуговуються, як для досягнення певних рішень, просто переглядаючи візуалізацію. Користувач повинен був висловити свою думку щодо різної візуалізації для одного запиту в рейтингах. Усі оцінки складалися з наданням зірок від 1 до 5. Також були кнопки «подобається» та «не подобається», щоб надати думку користувачів. Усім користувачам також було запропоновано висловити свою особливу думку щодо запитів, наприклад у формі коментарів. Коли користувачі входили в систему, їхній час також реєструвався, крім того, скільки часу користувач витратив на кожен запит, щоб прийняти певні рішення? Ці зареєстровані дані можуть бути використані для вивчення цілей.

Кожен запит показує різні результати. Ці результати, автоматизовані в графічній формі, надали користувачам чітке уявлення про набори даних щодо того, що потрібно для прийняття певних рішень. Були завдання, які вимагали від користувача вивчити різну візуалізацію та вибрати найкращу візуалізацію для даного запиту, що вже було зроблено для наборів даних. Користувачі також можуть надати свої коментарі та примітки, якщо рівень їхнього задоволення не був досягнутий.

Час початку записувався для кожного користувача, коли він/вона почав працювати з системою, а час завершення зазначався, як тільки користувач закінчив вибір усіх запитаних вимог та відкидав решту змінних вимог. Тривалість часу, який користувач витратив на виконання завдань із системою, розраховували для того, щоб дослідити, в середньому, скільки часу потрібно користувачам для виконання завдання. Цей час знадобиться для розрахунку ефективності системи, тобто про кількість завдань, які користувач може виконати за інтервал часу при запуску, і про те, як інтервал часу збільшується або зменшується в міру адаптації системи та один з головних факторів у тестуванні юзабіліті.

Маючи адаптивні та інші результати дослідження для адаптивної візуалізації даних, ці дві системи, одна з адаптованим шляхом, а інша без адаптивної, можна було б порівняти з точки зору ефективності та рівня помилок. Щоб розрахувати інші фактори в дослідженні юзабіліті, такі як здатність до навчання, ефективність та задоволеність користувачів, учасникам було запропоновано заповнити анкету, і відповідно до їхніх відповідей були проаналізовані інші фактори юзабіліті.

3.4 Час проектування рішення

Дослідження того, що робили користувачі від початку опитування до кінця опитування, було проведено для виміряного часу адаптованої рамки візуалізації даних до звичайної групи фреймворків візуалізації даних.

Для двох різних груп, однієї для інформатики, а іншої для студентів бізнес-класу, реєструвалися дані, щоб зафіксувати час, необхідний користувачам для виконання запитаного опитування. При запуску рівень адаптації системи був нульовим і за замовчуванням він показував однакові діаграми для всіх запитів. Користувачу знадобився час, щоб вибрати найкращі результати для запиту, але, як ми бачили, через певний проміжок часу, коли система показувала результати

адаптації в міру адаптації, вона відображала найкращий результат для даного запиту, тому час, витрачений на вирішення запиту, зменшувався.

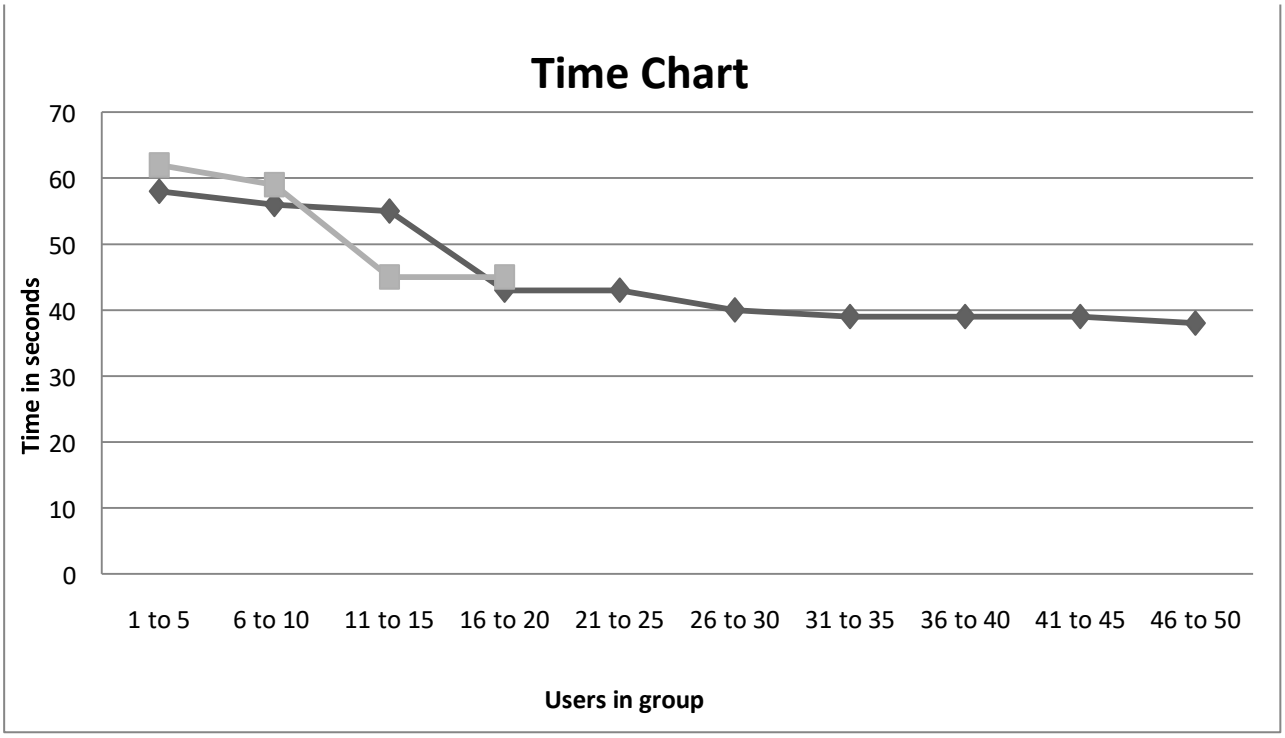


Рисунок 3.1 - Середній час виконання запиту

Тут, на рисунку 3.1, зображено середній час, необхідний для виконання запиту двома групами, тобто бізнесом та інформатикою. На осі X є групи запитів. Ми розбиваємо групи запитів у діапазоні від 1 до 5 і відкладаємо їх середній час, тоді як вісь Y описує час у інтервалі секунд. Таким чином, щоб вибрати найкращу візуалізацію, перша група користувачів із інформатики, тобто 1-5, зайняла 58 секунд, тобто середній час для всіх 5 користувачів, а друга група, тобто студенти бізнес-класу, зайняли 62 секунди. Усі графіки будуються за середнім часом кожної групи проти кожної групи.

Коли ми аналізуємо графік, ми бачимо, що на початку студентам з інформатики потрібно більше 50 секунд, але коли система адаптується і показує результати адаптованого шляху, користувач може побачити найкращі результати заданого обмеження часу запиту, яке зменшується з діапазону 50 секунд. до 40

секунд, і після певних запитів він залишається незмінним у діапазоні 35-40 секунд. У студентів бізнес-класу він починається з 62 секунд і зменшується до діапазону 45 секунд, однак для досягнення цього діапазону потрібно менше часу. Оскільки в цій групі лише 15 студентів, а студенти бізнес-класу краще входять до статистичного аналізу, ніж студенти інформатики

Таким чином, вищенаведені результати передбачають, що адаптована структура візуалізації даних вважається важливим інструментом у прийнятті рішень. Як ми бачили, з плином часу система стає все більш адаптованою і скорочує час на вирішення певних запитів.

3.5 Адаптація

Адаптація визначається як адаптований шлях системою. З плином часу ми бачимо, що всі системи адаптувалися. Під час тестування юзабіліті важливо отримати результати адаптованої системи, оскільки стартова система знаходиться в нейтральному стані, але з часом, коли користувач взаємодіє з системою, система сама фіксує певну поведінку, а також користувачі надають свій вхід до системи. Усі форми поведінки фіксуються в механізмі адаптації. Ми бачимо, що на даній діаграмі на початковому рівні адаптації є 0. Кожен користувач взаємодіє з системою і надає зворотний зв'язок. Діаграма займає перше місце, оскільки більшість користувачів вибирають візуалізацію бульбашкової діаграми найкраще для цього запиту. Також для цього запиту найкраще підходить друга діаграма, тобто гістограма. Коефіцієнт адаптації для бульбашкової діаграми становить близько 20, а для стовпчастої діаграми дорівнює 13, а для кругової та лінійної діаграми дорівнює 9. Коли адаптивні результати одного запиту порівнюються із запитами того самого типу, ми бачимо, що вподобання користувача отримують певний шлях, а система отримує адаптований, цим шляхом йдемо.

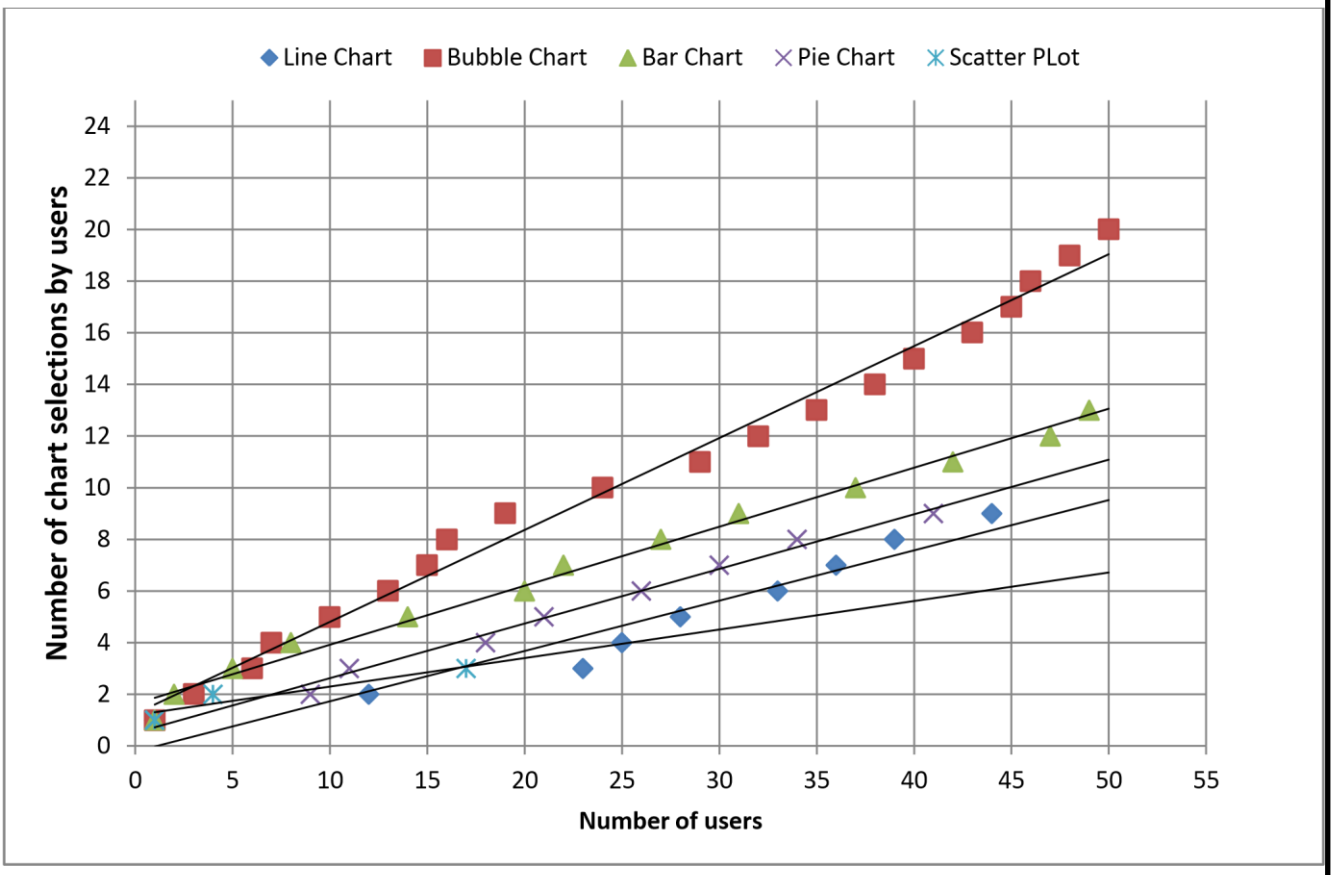


Рисунок 3.2 - Коефіцієнт адаптивності запиту

Це зручність використання забезпечує шлях для майбутніх розробок адаптивної системи. Оскільки всі дані реєструються, ці результати надалі використовуються для демонстрації та повторного використання цих результатів, щоб забезпечити користувачам найкращу візуалізацію конкретного домену в системах візуалізації.

Анкета

Питання анкети були розроблені для вирішення основних факторів зручності використання, якими були здатність до навчання, ефективність, рівень помилок та задоволеність. У цій частині обговорювалося кожне питання та мета їх розробки для оцінки кожного фактора та оцінювались усі результати, отримані з думок користувачів.

Здатність до навчання

Перші два запитання мали на меті оцінити здатність систем до навчання. Як зазначалося раніше, здатність до навчання полягає в тому, наскільки швидко і

легко користувачі можуть працювати з новою системою вперше. Цей фактор може вплинути на легкість системи на основі думки користувача, а також думки користувача про його/її розуміння системи.

Легкість

У першому питанні користувачам було запропоновано оцінити рівень простоти роботи з системою на основі свого досвіду. Відповідь користувачів на це запитання представляла їхнє особисте враження про досвід роботи з системою. На рисунку 3.3 видно, що середній бал для першої групи з інформатики, яку ми класифікували як перші 25 учасників, що становить 5,9 з 10, є нижчим за середній бал для другої групи, тобто 26-50 учасників, що становить 7,8 . Це також підтверджується вищим середнім балом, отриманим у другій бізнес-групі, який становить 6,72 і на 0,89 вище, ніж середній для першої групи 5,83. Це підтверджує, що в міру адаптації системи збільшувалася легкість, і користувач знайшов адаптовану систему легше.

На рисунку середні бали легкості були класифіковані для груп студентів бізнес-класу та студентів інформатики. Загалом середня оцінка легкості, надана групою B як для комп’ютерів, так і для студентів бізнес-класу, була вищою, ніж оцінка та запропонований рейтинг легкості всіма користувачами з групи A. Однак у групі A, хоча середній бал, отриманий з інформатики група, яка становить 5,9, вища за бал у групі студентів бізнес-класу, який становить 5,83, студенти з бізнесу дали нижчий бал легкості, ніж студенти з інформатики. Ця різниця може бути пов’язана або з неоднаковою кількістю учасників з кожної галузі, або через різницю в освіті учасників, що може надати їм різні стандарти для визначення легкості.

У випадку групи графічного інтерфейсу як середній бал, так і середній бал легкості для студента інформатики вищий, ніж для студентів бізнес-класу.

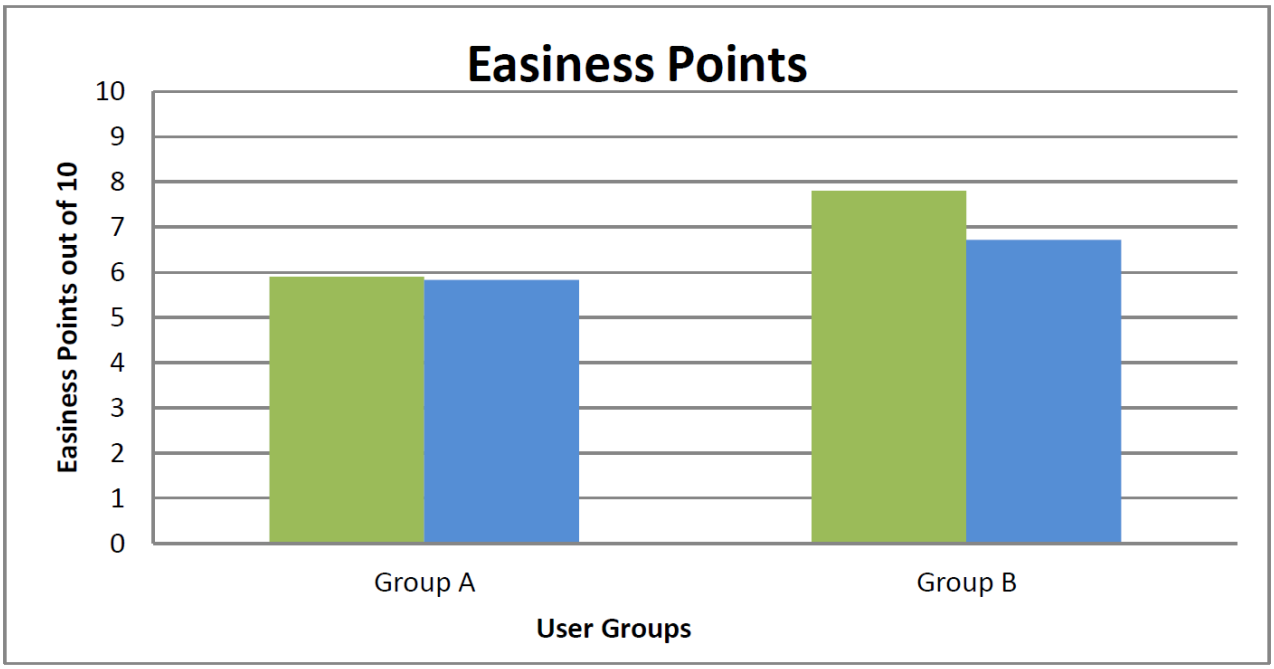


Рисунок 3.3 - Порівняння середньої точки легкості різних груп

Зрозумілість

Іншим фактором, який оцінювався в цьому дослідженні, є зрозумілість. Користувачам було запропоновано оцінити своє розуміння системи від 1 до 10. Оскільки цей бал був даний користувачам самостійно, нам потрібно було змінити цей рейтинг залежно від того, наскільки успішно вони виконували завдання та як швидко вони виконували завдання. Тому таку формулу (1), яка задовольняє ці потреби, запропонував Хань Х. Лін [21].

$$\text{Understanding(U)} = \frac{\text{User_rank} \times \text{Score}}{\text{Time}} \quad (1)$$

У цій формулі оцінка показує, наскільки успішними були користувачі у виконанні завдань. Таким чином, високий бал показує суб'єкта, який мав кращу продуктивність (вибрав і відмовився від правильних вимог за менший час) і краще зрозумів системи, і тому запропонований ним бал для його розуміння є більш достовірним і мав би більшу вагу, і навпаки.

На рисунку 3.4 проведено порівняння між значеннями зрозумілості та простоти. Можна помітити, що значення зрозумілості в групі B вище, ніж у групі A для обох груп користувачів з комп'ютерного відділу та бізнес-відділу.

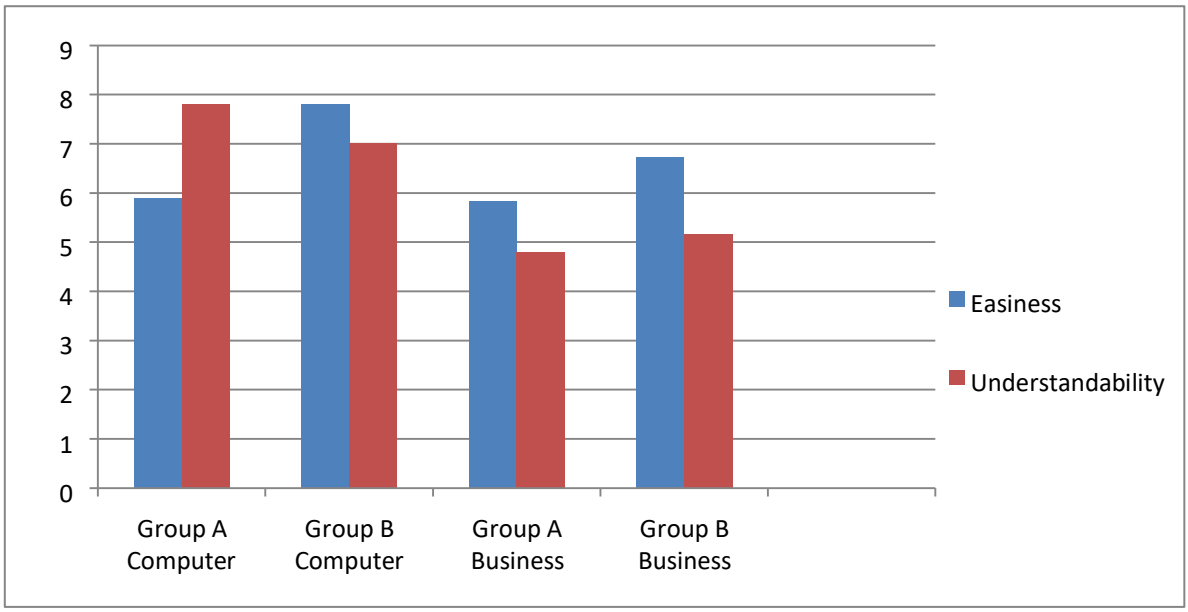


Рисунок 3.4 – Порівняння точки легкості з оціненою точкою зрозумілості

Крім того, як згадувалося раніше, коефіцієнт легкості для групи B був вищим, ніж для групи A. Оскільки і легкість, і зрозумілість для групи B були вищими, ніж для групи A, можна зробити висновок, що здатність до навчання групи B вища як для користувачів комп’ютерів, так і для користувачів комп’ютерів. бізнес-користувачі.

Таблиця 3.2 – Баланси легкості та розуміння для різних груп

	Group A Computer	Group B Computer	Group A Business	Group B Business
Easiness	5.9/10	7.8/10	5.83/10	6.72/10
Understandability	3.85 U	7.01 U	4.79 U	5.16 U

Ефективність

Як зазначалося раніше, коефіцієнт ефективності являє собою кількість завдань, які може виконати користувач за певну одиницю часу. Таким чином, ефективність можна розрахувати, поділивши бал на час, необхідний користувачам для виконання завдань. Слід зазначити, що оцінка розраховується на основі кількості завдань, які користувачі успішно виконали.

Як видно на рисунку, середня ефективність у групі B становить 0,78, тоді як середня ефективність у групі A становить 0,54. Іншими словами, ефективність у групі B приблизно на 0,24 вища, ніж ефективність у групі A, що означає, що за ту саму одиницю часу користувачі можуть виконувати на 40% більше завдань правильно, якщо користувачі використовують адаптовану систему, ніж звичайну.

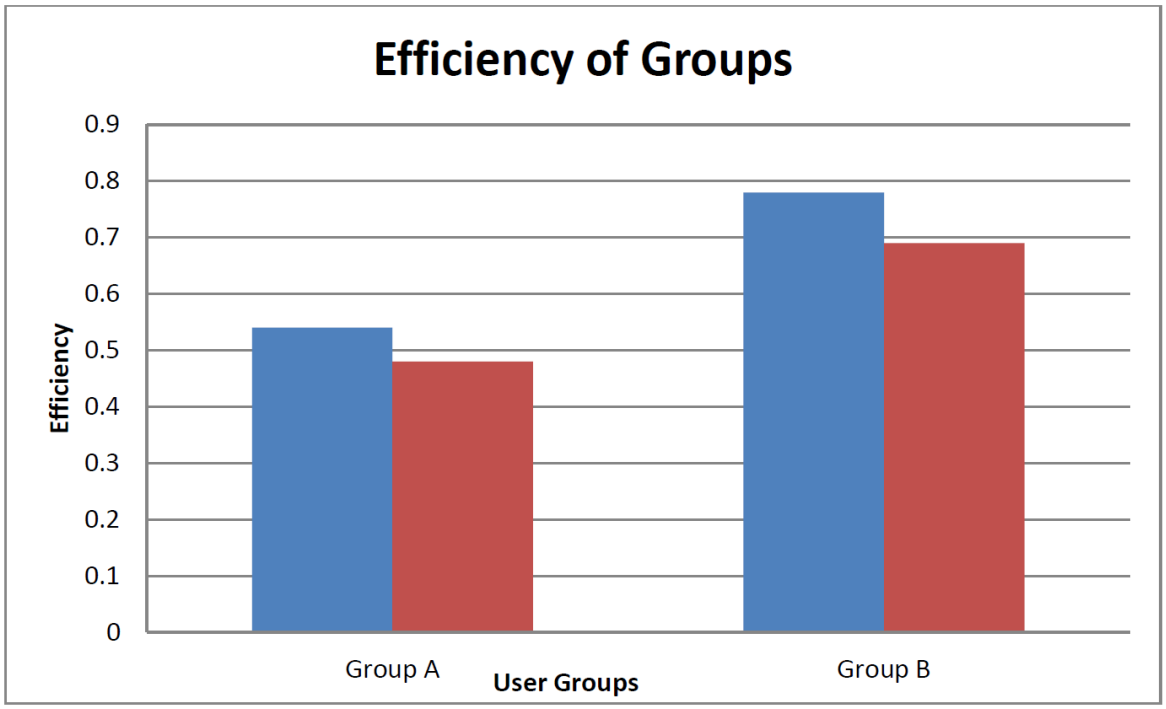


Рисунок 3.5 – Порівняння ефективності між різними групами

На рисунку 3.5 ефективність для різних користувачів на основі їхнього рівня знань у комп'ютерних питаннях розділена на чотири групи. Подивившись на Рисунок, можна побачити, що ефективність користувачів, які використовували

адаптовану систему, незалежно від того, чи були вони з комп'ютерного відділу чи бізнес-відділу, вища, ніж у користувачів, які використовували звичайний інтерфейс. Зокрема, ефективність комп'ютерних студентів, які використовували адаптовану систему, вища за інших користувачів.

Дивлячись на цей фактор, ми можемо зробити висновок, що наявність адаптованої системи візуалізації даних допомагає користувачам працювати з системою швидше та точніше.

Задоволеність користувачів

Останнім фактором зручності використання, який оцінювався в дослідженні, було задоволеність користувачів. Одне із запитань в анкетах було розроблено для оцінки загальної задоволеності користувача роботою з кожною системою. У цьому питанні всім учасникам було запропоновано оцінити загальну задоволеність системою з 10. Результати були такими.

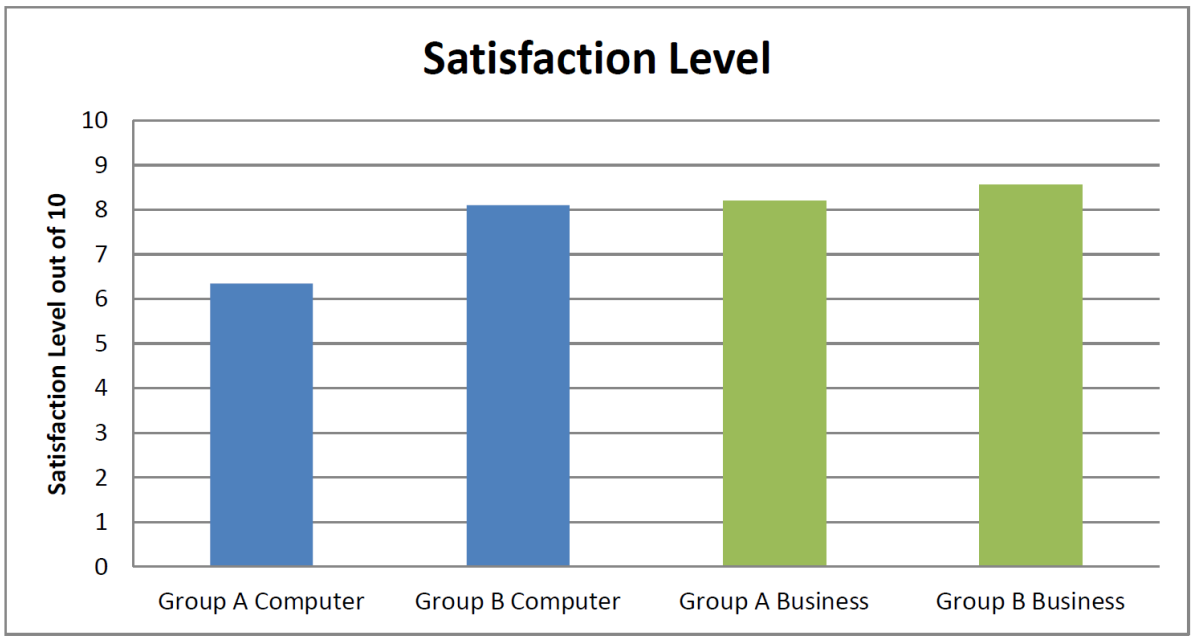


Рисунок 3.6 – Загальна задоволеність користувачів системою для різних груп

Середні бали групи А учнів комп'ютера становили 6,35. Цей бал збільшується до 8,2 для студентів бізнес-класу, які працювали з тією ж системою.

Для системи з групою В студенти-комп'ютерники були задоволені на 8,1, а студенти з бізнес-класу – 8,55. Ця статистика проілюстрована на рисунку 6.6. Загалом усі користувачі системи групи А поставили оцінку 7,27 на своє задоволення системою, а користувачі системи групи В визначили свій рівень з оцінкою 8,32 з 10.

При порівнянні двох систем загалом учасники групи Б, тобто системи адаптованої візуалізації, були задоволені більше, ніж учасники групи А, тобто класичної системи візуалізації. У середньому оцінка задоволеності користувачів IT-студентами, які працювали з адаптованою системою, була вищою, ніж студентами бізнесу, які працювали з неадаптованою системою.

Щоб продемонструвати твердження, що обрана адаптована система візуалізації даних покращує інтерфейс простої системи візуалізації різними способами, крім питань про фактори зручності, в анкети були додані деякі інші запитання, наприклад, думки користувачів про те, що зміни можуть бути внесені в кожну систему та необхідність візуалізації для них тощо.

Необхідність

Одним із важливих запитань було «Чи вважаєте ви, що адаптована візуалізація потрібна для фреймворку візуалізації?» Це питання задавали лише студентам, які працювали з адаптованою візуалізованою системою. В основному, це питання може показати, чи користувачі вважають адаптовану систему корисною, а адаптивність підвищує їх ефективність. Варіанти відповідей на це запитання були: необхідно, без різниці і не потрібно. Загалом 77% студентів вважають, що це необхідно, і 13% студентів вважають, що не має значення, адаптовано це чи ні, і лише 10% вважають, що адаптація не потрібна.

Отже, зрозуміло, що більше студентів вважають, що система має бути необхідною, а адаптація в цілому покращує їхню продуктивність та розуміння.

Проблеми

В анкеті є запитання з кількома відповідями для студентів, які працювали з цією системою, що дає користувачам можливість додати свою думку щодо

проблем, які має система. Три параметри, які були дуже помітні, засновані на стандартах зручності розробки інтерфейсу користувача, були надані користувачам як варіанти, а також їх попросили додати власну думку, якщо вони знайдуть більше проблем у системі. Їм були запропоновані наступні варіанти:

(a) Потрібні додаткові методи візуалізації даних та атрибути.

(b) Неправильна візуалізація даних заданих запитів і незрозуміла.

(c) Система не в адаптованому стані.

(d) Трендовий перегляд необхідних даних.

Між цими трьома варіантами відсоток студентів з комп'ютера групи А, які вибрали варіант «а», становив 30%. 20% погодилися на вибір «b», а 56% – на вибір «в» і 38 %. Серед студентів-бізнесменів вибір «а» обрали 24% учасників, вибір «б» — 20%, вибір «в» — 48% і вибір «г» — 48% з них.

Для групи В, серед цих трьох варіантів, відсоток студентів з комп'ютерної групи В, які вибрали варіант «а», становив 32%. 19% погодилися з вибором «b», а 26% з варіантом «с» і 32 %. Серед студентів-бізнесменів вибір «а» обрали 22% учасників, вибір «б» — 23%, вибір «в» — 34% і вибір «г» — 43% з них.

Деякі учасники додали власні ідеї щодо системи. Деякі з доданих коментарів були такими:

• Можливість відображати адаптований результат чи ні?

• Можливість переглянути докладнішу інформацію про дані.

• Плутанина в деяких значеннях даних.

• Повинен мати опцію аудіо.

• Плавна візуалізація.

ВИСНОВКИ

В дипломній роботі було проведено аналіз, щоб зробити висновок щодо адаптованої техніки візуалізації даних і подальших вдосконалень, що застосовуються до простої системи візуалізації даних, це дає користувачам краще розуміння наборів даних і зменшує час і зусилля, які їм потрібно витрачати. щодо виявлення бажаних вимог. Це дослідження було виконано в кілька етапів. Спочатку була вивчена попередня доступна адаптована система та система візуалізації, і на основі базових концепцій проектування цієї системи було розроблено та впроваджено найбільш підходящу адаптовану систему візуалізації даних як з точки зору семантики, так і з точки зору зручності використання. Крім того, було проведено дослідження зручності використання на групі студентів з різним академічним досвідом, щоб підтвердити, що запропонований дизайн може позитивно покращити зручність використання адаптованої системи візуалізації даних.

Результати дослідження показують, що в цілому користувачі позитивно ставляться до використання адаптивної візуалізації даних. Коли оцінюються певні параметри та певні результати на основі тестування юзабіліті фреймворка, а саме:

1. Запропоновано концепцію впровадження підходу адаптивності, щоб збагатити структуру візуалізації даних, яка може надати низку візуалізацій даних, що найкраще підходять для заданої якості даних. Наш фреймворк здатний виробляти результати адаптивної візуалізації, які впроваджуються та тестуються.

2. Фреймворк здатний підтримувати багатомовне розгортання і може давати однаково якісні результати на всіх платформах.

3. Проведено тест на зручність використання фреймворка і таким чином створено еталон для порівняння з майбутніми покращеннями.

СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА

1 J. L. V. Sancho, J. C. Dominguez, and B. E. M. Ochoa, “An approach to the taxonomy of data visualization,” Revista Latina de Communicacion Social, vol. 69, 2014, pp. 486-507.

2 “Data visualization techniques,” SAS, html?keyword=data+visualization+techniques&matchtype=p&publisher=google&gclid=COycgoCbutACFcolgQodqwgIiA J. Wolfe, “Teaching students to focus on the data in data visualization,” Journal of Business and Technical Communication, vol. 29, no. 3, 2015, pp. 344-359.

3 T. Kilimba, G. Nimako, and K. Herbst, “Data everywhere: an integrated longitudinal data visualization platform for health and demographic surveillance sites,” Proceedings of the 6th ACM

4 Conference on Bioinformatics, Computational Biology and Health Informatics, Sept. 2015, pp. 551,552.

5 O. Kumar and A. Goyal, “Visualization: a novel approach for big data analytics,” Proceedings of the Second International Conference on Computational Intelligence & Communication Technology, 2016, pp. 121-124.

6 S. Grainger, F. Mao, and W. Buytaert, “Environmental data visualization for non-scientific contexts: Literature review and design framework,” Environmental Modelling and Software, vol. 85, 2016, pp. 299-318.

7 W. N. Dilla, R. L. Raschke, “Data visualization for fraud detection: Practice implications and a call for future research,” International Journal of Accounting Information Systems, vol. 16, 2015, pp. 1-22.

8 S. A. Murhy, “Data visualization and rapid analytics: applying tableau desktop to support library decision-making,” Journal of Web Librarianship, vol. 7, no. 4, 2013, pp. 465-476.

9 T. J. Brigham, “Feast for the eyes: an introduction to data visualization,” Medical Reference Services Quarterly, vol. 35, no. 2, 2016, pp. 215-223.

10 C. Chen, “Information visualization,” WIREs Computational Statistics, vol. 2, July/August, 2010, pp. 387-403.
11 R. R. Laher, “Thoth: software for data visualization and statistics,” Astronomy and Computing, vol. 17, 2016, pp. 177-185.
12 L. Yu et al., “Automatic animation for time-varying data visualization,” Pacific Graphics, vol. 29, no. 7, 2010, pp. 2271 -2280.
13 X. Li et al., “Advanced aggregate computation for large data visualization,” Proceedings of IEEE Symposium on Large Data Analysis and Visualization, 2015, pp. 137,138.
14 Outsourcing and offshoring in the 21st century. Harbhanjan s.kehal and varinde p.singh. 2006, ISBN 9781591408772, e-library.
15 THE FUTURE OF IT OUTSOURCING: 6 TRENDS FOR 2019. URL: https://devabit.com/blog/it-outsourcing-trends-2019.
16 Black Book of outsourcing. How to manage the changes, challenges and opportunities” Brown, Douglas, Wilson, Scott. 2005. ISBN: 9780471718895, ebrary.
17 The Process of outsourcing. Strategic and operational realities. Reine Wasner. 1999.
18 Полуектова Н., Книшенко Т. Методи прийняття рішень щодо аутсорсингу сервісів інформаційної системи підприємства. Сумський державний університет. Маркетинг і менеджмент інновацій. 2015. No 1
19 D.Dirlav. (2007), Izbrannye koncepcii biznesa. Teorii, kotorye izmenili mir [Selected business concept. The theories that changed the world], Olimp-Biznes.
20 Eliferov V.G., Repin V.V. (2007), Bizness-processy: Reglamentacija i upravlenie, [Business processes: Regulation and Management], INFRA-M.
21 J.Bekker (2008), Menedzhment processov [Process management], Jeksmo.
22 Fedonin O.S., Shvidanenko G.O., Lavrenenko V.V. (2011), Novitni tendencii rozvitku upravlinnja pidpriємstvami [The latest trend of business management], KNEU,. Kyiv, Ukraine.

23 Binner H. (2010), Upravlenie organizacijami i proizvodstvom: Ot funkcional'nogo menedzhmenta k processnomu [Organization and management: from a functional management to the process].

24 Tebekin A.V. (2013), Metody prinjatija upravlencheskih reshenij [Methods for making management decisions].

25 Uchitel' Ju.G. (2007), Razrabotka upravlencheskih reshenij: uchebnik dlja studentov vuzov, obuchajushhihsja po special'nosti "Antikrizisnoe upravlenie i drugim jekonomicheskim special'nostjam, special'nosti "Menedzhment organizacii" [Development of managerial decisions], 2nd ed, JuNITI.

26 Sitnik V.F. (2004), Sistemi pidtrimki prijnjattja rishen' [Decision support systems], KNEU,. Kyiv, Ukraine.

27 Oleksjuk O.S. (1998), Sistemi pidtrimki prijnjattja finansovih rishen' na mikrorivni [Decision support financial systems at the micro level], Naukova dumka, Kyiv, Ukraine.

28 Pomazun O.M. (2015), "Features of decision making in business process management", Instrumenty rehuliuvannia natsional'noi ekonomiky v umovakh suchasnykh hlobal'nykh vyklykiv: materialy Mizhnarodnoi naukovo-praktychnoi konferentsii [Tools regulation of national economy in today's global challenges: materials of the International Scientific Conference], Khmel'nyts'kyj, Ukraine, pp.64-66.

БІБЛІОГРАФІЧНА ДОВІДКА

Тема дипломної роботи бакалавра: "Інформаційні моделі візуалізації процесів прийняття рішень"

Обсяг пояснювальної записки: 66 аркушів

Дата закінчення дипломної роботи 10 червня 2022р.

Підпис студента _____