Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет           навчально-науковий центр заочної форми навчання

(повна назва)

Кафедра програмної інженерії

(повна назва)

**СПЕЦИФІКАЦІЯ**

**програмного продукту до**

**кваліфікаційної роботи**

рівень вищої освіти другий (магістерський)

Дослідження методів ієрархії моделей знань предметної області

(тема)

Виконала:

здобувач 2 року навчання

групи ІПЗзм-23-1

Аліна ЗАГОВОРА

(Власне ім’я, ПРІЗВИЩЕ)

Спеціальність 121 – Інженерія програмного

забезпечення

(код і повна назва спеціальності)

Тип програми освітньо-наукова

Керівник доц. Олексій НАЗАРОВ

(посада, Власне ім’я, ПРІЗВИЩЕ)

2025 р.

**1. Назва програмного продукту:**

AHP vs ANP: моделювання ієрархії знань предметної області (вибір IT – спеціальності).

**2. Призначення програмного продукту:**

Програмне забезпечення розроблено для демонстрації методів прийняття рішень на основі ієрархії моделей знань предметної області, зокрема Аналітичного Ієрархічного Процесу (AHP) та Аналітичного Мережевого Процесу (ANP). Воно призначене для дослідницьких цілей, забезпечуючи розрахунок ваг критеріїв та альтернатив, оцінку їх узгодженості та генерацію звітів із результатами аналізу.

**3. Короткий опис функціональних можливостей**

1. Розрахунок ваг критеріїв (AHP):
   * обробка матриць попарних порівнянь для визначення ваг критеріїв;
   * розрахунок індексу узгодженості (CI) та коефіцієнта узгодженості (CR);
   * вимірювання часу виконання обчислень.
2. Оцінка альтернатив (AHP):
   * розрахунок локальних ваг альтернатив для кожного критерію;
   * обчислення глобальних ваг альтернатив на основі ваг критеріїв;
3. Розрахунок ваг критеріїв та альтернатив (ANP):
   * побудова суперматриці з урахуванням залежностей між критеріями;
   * розрахунок пріоритетного вектора для критеріїв та альтернатив;
   * обробка залежностей та впливів між елементами моделі;
   * оцінка узгодженості матриць порівнянь.
4. REST API:
   * інтерфейс для інтеграції з зовнішніми системами;
   * ендпоінти для виконання обчислень AHP, ANP та оцінки альтернатив.

**4. Таблиця з ендпоінтами REST API**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ендпоінт | Метод | Опис | Вхідні дані | Вихідні дані |
| /calculate | POST | Розрахунок ваг критеріїв за AHP та оцінка узгодженості матриці порівнянь. | **JSON: { "matrix": [[...]] }** | **JSON: { "weights": [...], "ci": number, "cr": number, "execution\_time": number }** |
| /final-calculate | POST | Розрахунок глобальних ваг альтернатив за AHP. | **JSON: { "criteria\_weights": [...], "alternative\_matrices": { "crit1": [[...]], ... } }** | **JSON: { "final\_scores": [...], "consistency": {...}, "alternative\_weights": {...}, "execution\_time": number }** |
| /api/anp | POST | Розрахунок пріоритетів критеріїв за ANP з урахуванням залежностей. | **JSON: { "criteria": [...], "dependencies": {...}, "comparisons": [[...]], "influence": {...} }** | **JSON: { "supermatrix": [[...]], "priority\_vector": {...}, "ci": number, "cr": number, "execution\_time": number }** |
| /anp\_alternatives/consistency | POST | Оцінка узгодженості матриці порівнянь для альтернатив (ANP). | **JSON: { "matrix": [[...]] }** | **JSON: { "ci": number, "cr": number }** |
| /anp\_alternatives/final | POST | Розрахунок фінальних ваг альтернатив за ANP. | **JSON: { "criteria": [...], "matrices": {...}, "criteria\_weights": {...}, "dependencies": {...}, "influence": {...} }** | **JSON: { "final\_scores": [...], "consistency": {...}, "alternative\_weights": {...}, "execution\_time": number }** |

1. **Технології та інструменти**

**Мова програмування:**

* Python 3.8+

**Фреймворки та бібліотеки:**

* Flask: для створення REST API.
* NumPy: для матричних обчислень.
* Flask-CORS: для обробки CORS-запитів.

**Інструменти для розробки:**

* + IDE: VS Code.
  + Git: для контролю версій.

**Інструменти для тестування:**

* Postman: для тестування REST API.

**Залежності:**

* flask, numpy, flask-cors.

1. **Цільова платформа**

Операційна система: Кросплатформне рішення, сумісне з Windows, macOS, Linux.

Середовище виконання: Локальний сервер Flask для розробки та тестування; можливе розгортання на веб-сервері (наприклад, Gunicorn + Nginx).

Клієнтська частина: Веб-браузер або будь-який клієнт, здатний виконувати HTTP-запити (наприклад, Postman, cURL).

Формат даних: JSON для API-запитів.

**7. Обмеження та припущення**

**Обмеження:**

* матриці попарних порівнянь мають бути квадратними та позитивними;
* розмірність матриць обмежена таблицею RI (до 10 елементів);
* програма не підтримує асинхронну обробку великих обсягів даних;

**Припущення:**

* вхідні матриці попарних порівнянь є коректними (без від’ємних значень);
* вхідні дані для API надходять у правильному форматі JSON;
* сервер має достатньо ресурсів для виконання матричних обчислень;
* залежності та впливи в ANP надаються у коректному форматі.

**8. Критерії якості**

Точність: Результати обчислень (ваги, CI, CR) відповідають математичним формулам AHP та ANP з точністю до 4 знаків після коми.

Продуктивність: Час виконання обчислень не перевищує 1 секунди для матриць розміром до 10x10.

Надійність: Програма обробляє помилкові вхідні дані, повертаючи відповідні повідомлення про помилки (HTTP 400/500).

Зручність використання: API ендпоінти мають чітку документацію та передбачувану поведінку.

Сумісність: Сумісність із сучасними версіями Python (3.8+) та браузерами для доступу до API.