|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА - Российский технологический университет»**  **РТУ МИРЭА** |

Институт Информационных Технологий

Кафедра математического обеспечения и стандартизации информационных технологий (МОСИТ)

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ** **№1**

по дисциплине «Моделирование программных систем»

Выполнили студенты группы ИКБО-11-22 Андрусенко Л.Д.

Гришин А.В.

Работа выполнена «\_\_» \_\_\_\_\_\_\_202\_\_ г

«Зачтено» «\_\_» \_\_\_\_\_\_\_202\_\_ г.

Москва 2024

**Содержание**

[Введение 3](#_Toc129621334)

[Постановка задачи 4](#_Toc129621335)

[Ход работы 5](#_Toc129621336)

[1. Выбор критериев оценки и построение таблицы оценки ПО 5](#_Toc129621337)

[2. Выбор и реализация методов 6](#_Toc129621338)

[2.1. Метод ELECTRE 6](#_Toc129621339)

[2.2. Метод TOPSIS 10](#_Toc129621340)

[2.3. Метод SAW 13](#_Toc129621341)

[3. Выбор оптимального ПО 15](#_Toc129621342)

[Вывод 16](#_Toc129621343)

[Список источников 17](#_Toc129621344)

# Введение

Оценка качества программного обеспечения на сегодняшний день является важным этапом при выборе ПО для профессионального использования. В данной работе были выбраны мессенджеры Signal, ICQ и WeChat для проведения оценки качества. Были определены критерии оценки, построена таблица оценки и применены методы принятия управленческих решений ELECTRE, TOPSIS, SAW для выбора оптимального варианта программного обеспечения.

# Постановка задачи

Провести оценку качества программного обеспечения и осуществить обоснованный выбор варианта программного обеспечения, применяемого в профессиональной сфере в соответствии с указанной методикой.

В качестве программного обеспечения были выбраны мессенджеры (Signal, ICQ, WeChat), а в качестве методов принятия управленческих решений были выбраны методы ELECTRE, TOPSIS, SAW.

# Ход работы

## Выбор критериев оценки и построение таблицы оценки ПО

В качестве критериев оценки программного обеспечения выберем следующие:

* Защищённость данных;
* Удобство использования;
* Возможности коммуникации;
* Надёжность работы;
* Мобильность (Возможность использования приложения на различных устройствах);

Далее построим таблицу с оцениваемыми мессенджерами в сроках и с критериями в столбцах в нормализованном виде, а также установим веса для каждого критерия:

Таблица 1. Оценка ПО

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Веса** | 0,3 | 0,25 | 0,2 | 0,15 | 0,1 |
| **Характе- ристики**  **Маркет-**  **плейсы** | Защищённость данных | Удобство использования | Возможности коммуникации | Надёжность работы | Мобильность |
| Signal | 0,9 | 0,85 | 0,75 | 0,85 | 0,95 |
| ICQ | 0,7 | 0,75 | 0,85 | 0,8 | 0,9 |
| WeChat | 0,8 | 0,9 | 0,9 | 0,75 | 0,85 |

## Выбор и реализация методов

### Метод ELECTRE

Методы ELECTRE предоставляют возможность оценить степень превосходства одного альтернативного решения над другими с помощью анализа их согласования. Процесс принятия решения начинается с раунда оценивания, когда эксперт оценивает все альтернативные решения по всем критериям. В результате формируется матрица решений 𝐴 = (𝑥𝑖𝑗), где 𝑥𝑖𝑗 обозначает оценку, данную по 𝑖-той альтернативе по 𝑗-тому критерию. Метод состоит из 9 последовательных шагов:

* расчёт нормализованной матрицы решения;
* расчёт взвешенной нормализованной матрицы решения;
* построение множеств согласия и несогласия;
* расчёт матрицы согласия;
* расчёт матрицы несогласия;
* построение матрицы индексов согласия превосходства;
* построение матрицы индексов несогласия превосходства;
* построение агрегированной матрицы превосходства;
* удаление наименее предпочтительных альтернатив.

На основе этого напишем код для подсчёта и получения ранжированного списка анализируемого ПО (Листинг 1).

Листинг 1. Метод ELECTRE

|  |
| --- |
| def electre(weights, characteristics, threshold):  num\_alternatives = len(characteristics)concordance\_matrix = [[0] \* num\_alternatives for \_ in range(num\_alternatives)]  for i in range(num\_alternatives):  for j in range(num\_alternatives):  concordance\_matrix[i][j] = sum(  weights[k] \* (characteristics[j][k] >= characteristics[i][k]) for k in range(len(weights))  ) / sum(weights)dominance\_matrix = [[int(concordance\_matrix[i][j] >= threshold) for j in range(num\_alternatives)] for i in  range(num\_alternatives)]nondominated\_alternatives = [i for i in range(num\_alternatives) if sum(dominance\_matrix[i]) == num\_alternatives]  return dominance\_matrix, nondominated\_alternatives |

Передадим данные в нашу программу и получим результат, как показано ниже:

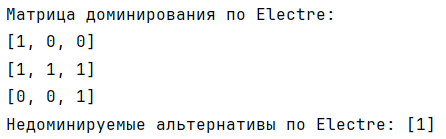


Рисунок 1 – Результат метода ELECTRE

Исходя из матрицы, можно сделать вывод, что по методу ELECTRE доминирует ICQ при пороге конкордации 0,6.

### Метод TOPSIS

TOPSIS – технология, разработанная Хвонгом и Юном в 1981 г. Данный метод используется для решения многокритериальных задач. Суть метода состоит в поиске альтернатив, значения оценок которых наиболее близки к идеально-позитивному решению и наиболее отдалены от идеально-негативного решения. Идеально-позитивное решение представляет собой вектор максимальных значений матрицы взвешенных оценок альтернатив. Идеально-негативное решение, напротив, является вектором минимальных значений.

Метод состоит из 6 последовательных шагов:

* расчёт нормализованной матрицы решения;
* расчёт взвешенной нормализованной матрицы решения;
* определение «идеального» и «идеально-негативного» ожидаемого состояния;
* расчёт метрики разделения;
* расчёт относительной близости к «идеальному» состоянию;
* ранжирование критериев.

На основе этого напишем код для подсчёта и получения ранжированного списка анализируемого ПО (Листинг 2).

Листинг 2. Метод TOPSIS

|  |
| --- |
| def topsis(weights, characteristics):  num\_alternatives, num\_criteria = characteristics.shape norm\_characteristics = characteristics / np.sqrt(np.sum(characteristics \*\* 2, axis=0)) weighted\_norm\_characteristics = norm\_characteristics \* weights ideal\_best = np.max(weighted\_norm\_characteristics, axis=0)  ideal\_worst = np.min(weighted\_norm\_characteristics, axis=0) ideal\_best\_scores = np.sqrt(np.sum((weighted\_norm\_characteristics - ideal\_best) \*\* 2, axis=1))  ideal\_worst\_scores = np.sqrt(np.sum((weighted\_norm\_characteristics - ideal\_worst) \*\* 2, axis=1)) scores\_topsis = ideal\_worst\_scores / (ideal\_worst\_scores + ideal\_best\_scores)   return scores\_topsis |

Передадим данные в нашу программу и получим результат, как показано ниже:



Рисунок 2 – Результат метода TOPSIS

В итоге получилось, что ICQ на 1-ом месте, WeChat на 2-ом, Signal на 3-ем.

### Метод SAW

Метод SAW, или метод простого аддитивного взвешивания, является одним из самых известных и широко используемых методов многоатрибутивного принятия решений. В целом процесс нахождения наилучшего поставщика может быть разделен на следующие этапы:

* анализ по критериям;
* определение весов критериев;
* нормирование критериев;
* определение рейтинга путем умножения значений критериев на веса.

На основе полученного рейтинга принимается решение о выборе. Однако результат оценки может иметь субъективный и неоднозначный характер, т.к. определение критериев оценки и присвоение удельной значимости факторам носят неформализованный характер и зависят от конкретных ситуаций. В зависимости от выбираемых критериев оценки наиболее предпочтительный выбор может меняться. Чаще всего для определения рейтинга используется экспертная оценка показателей с использованием балльной шкалы. Критерии могут оцениваться по пятибалльной, либо десятибалльной шкале. Критерии выбора, веса и оценки определяет менеджер по логистике, после чего происходит подсчет суммарного рейтинга.

На основе этого напишем код для подсчёта и получения ранжированного списка анализируемого ПО (Листинг 3).

Листинг 3. Метод SAW

|  |
| --- |
| def simple\_additive\_weighting(weights, characteristics):  scores = np.sum(weights \* characteristics, axis=1)  return scores |

Передадим данные в нашу программу и получим результат, как показано ниже:



Рисунок 3 – Результат метода SAW

В итоге получилось, что Signal на 1-ом месте, WeChat на 2-ом, ICQ на 3-ем.

## Выбор оптимального ПО

На основе результатов методов ELECTRE, TOPSIS и SAW получается, что 1-ое место получило ПО “ICQ”, а второе место делят ПО “Signal” и “WeChat”. Из чего можно сделать вывод, что ПО “ICQ” будет самым оптимальным выбором из просмотренных, за счет доминирующих позиций в удобстве пользования и защищенности.

# Вывод

В ходе выполнения практической работы мы научились:

* Оценивать качество программного обеспечения с использованием различных методов, таких как ELECTRE, TOPSIS и SAW.
* Проводить анализ критериев оценки программного обеспечения и определять их веса для построения обоснованных решений.
* Применять методы принятия управленческих решений для выбора оптимального варианта программного обеспечения.
* Понимать важность оценки качества программного обеспечения для повышения эффективности его использования в профессиональной сфере.
* Проводить анализ и обоснованный выбор программного обеспечения в соответствии с поставленными задачами и критериями оценки.

# Список источников

1. А. В. Демидовский. Сравнительный анализ методов многокритериального принятия решений: ELECTRE, TOPSIS и ML-LDM [Электронный ресурс]: Электронная статья: Национальный Исследовательский Университет Высшая Школа Экономики, 2020. – режим доступа: URL: <https://scm.etu.ru/assets/files/2020/scm20/papers/4/234.pdf>, свободный (дата обращения 06.03.2024)
2. Катаржина Халицкая. Выбор технологий с помощью метода TOPSIS [Электронный ресурс]: Электронная статья: Белостокский технический университет, 2020. – режим доступа: URL: <https://foresight-journal.hse.ru/data/2020/03/20/1567702093/6-Халицкая-85-96.pdf>, свободный (дата обращения 06.03.2024)
3. Леонас Устинович, Зенонас Турскис, Галина Шевченко. ПРИМЕНЕНИЕ МЕТОДА SAW ДЛЯ МНОГОКРИТЕРИАЛЬНОГО СРАВНИТЕЛЬНОГО АНАЛИЗА ВАРИАНТОВ РИСКА ИНВЕСТИЦИЙ В СТРОИТЕЛЬСТВЕ [Электронный ресурс]: Электронная статья: Вильнюсский технический университет им. Гедиминаса, 2006. – режим доступа: URL: <https://www.tsi.lv/sites/default/files/editor/science/Research_journals/Tr_Tel/2006/V3/art06-2.pdf>, свободный (дата обращения 06.03.2024)