БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ

ИНСТИТУТ ПО ИНФОРМАЦИОННИ И КОМУНИКАЦИОННИ ТЕХНОЛОГИИ

СЕКЦИЯ „ИНФОРМАЦИОННИ ПРОЦЕСИ И СИСТЕМИ ЗА ВЗЕМАНЕ НА РЕШЕНИЯ“

Зорница Агресимова Димитрова

МОДЕЛИ И СОФТУЕРНИ АРХИТЕКТУРИ НА СИСТЕМИ ЗА ПОДПОМАГАНЕ ВЗЕМАНЕТО НА РЕШЕНИЯ

ДИСЕРТАЦИЯ

за присъждане на образователна и научна степен „Доктор“

Професионално направление: 4.6. „Информатика и компютърни науки“

Докторска програма „Информатика“

Научен ръководител

проф. д.н. Даниела Борисова

София, 2025

СЪДЪРЖАНИЕ

Увод 5

Глава 1. Анализ на методите за вземане на решения и софтуерните архитектури за реализиране на уеб приложения 8

1.1. Методи за подпомагане вземането на решения 9

1.1.1. Концепции и класификации на методите за многокритериален анализ 9

1.1.2. Метод на претеглената сума и метод на претегленото произведение..... 12

1.1.3. Алгоритъм за вземане на решение чрез метода на претеглената сума и метода на претегленото произведение 15

1.1.4. Сравнение на метода на претеглената сума и метода на претегленото произведение 19

1.1.5. Модификации над метода на претеглената сума и метода на претегленото произведение 21

1.1.6. Системи за подпомагане вземането на решения..... 23

1.2. Софтуерни архитектури за реализиране на уеб приложения 25

1.2.1. Концепции на софтуерни архитектури, архитектурни стилове и шаблони ... 25

1.2.2. Софтуерни архитектурни стилове и шаблони..... 26

1.2.3. Характеристики за сравнение на софтуерни архитектури 31

1.3. Изводи 33

1.4. Цел и задачи 34

Глава 2. Модификации на метода на претеглената сума и метода на претегленото произведение 35

2.1. Модификация на метода на претеглената сума и метода на претегленото произведение, отчитаща различните области на компетентност на ЛВР 36

2.1.1. Алгоритъм за реализиране на модификацията, отчитаща различията в областите на компетентност на ЛВР 37

2.1.2. Математическа формулировка на модифицираните методи, отчитащи различията в областите на компетентност на ЛВР..... 39

2.2. Модификация на метода на претеглената сума, формализираща генериране на оценки за алтернативите 44

2.2.1. Алгоритъм за реализиране на модификацията, формализираща генериране на оценки за алтернативите..... 45

2.2.2. Математическа формулировка на модифицирания метод, формализиращ генериране на оценки за алтернативите 47

2.3. Модификация на метода на претеглената сума и метода на претегленото произведение, формализираща генериране на коефициенти, даващи предимство в общото представяне на алтернативите..... 50

2.3.1. Алгоритъм за реализиране на модификацията, формализираща генериране на коефициенти, даващи предимство в общото представяне на алтернативите .. 51

2.3.2. Математическа формулировка на модифицираните методи, формализиращи генериране на коефициенти, даващи предимство в общото представяне на алтернативите 54

2.4. Изводи 57

Глава 3. Проектиране на софтуерни архитектури на системи за подпомагане вземането на решения 58

3.1. Математически модели на комбинираните модификации на метода на претеглената сума..... 58

3.2. Обобщен алгоритъм, реализиращ комбиниране на модификациите 61

3.3. Проектиране на функционалността на СПВР..... 68

3.4. Уеб базирана СПВР с монолитна трислойна архитектура..... 72

3.5. Уеб базирана СПВР с безсървърна архитектура..... 75

3.6. Сравнение на предложените софтуерни архитектури..... 77

3.7. Изводи 79

Глава 4. Числено тестване на модифицираните методи 80

4.1. Числено тестване на модификацията, отчитаща компетентността на ЛВР в различни области на компетентност..... 80

4.1.1. Сценарии за тестване 81

4.1.2. Формулиране на задачата за числено тестване 82

4.1.3. Анализ на входните данни 86

4.1.4. Резултати от численото тестване на модификацията на метода на претеглената сума..... 89

4.1.5. Резултати от численото тестване на модификацията на метода на претегленото произведение..... 94

4.2. Числено тестване на модификацията, формализираща генериране на оценки за алтернативите 98

4.2.1. Сценарии за тестване 99

4.2.2. Формулиране на задачата за числено тестване 99

4.2.3. Резултати от численото тестване на модификацията на метода на претеглената сума 104

4.2.4. Резултати от численото тестване на комбинирания модел на метода на претеглената сума 110

4.3. Числено тестване на модификацията, формализираща генериране на коефициенти, даващи предимство в общото представяне на алтернативите 112

4.3.1. Сценарии за тестване 113

4.3.2. Формулиране на задачата за числено тестване 113

4.3.3. Резултати от численото тестване на модификацията на метода на претеглената сума 117

4.3.4. Резултати от численото тестване на модификацията на метода на претегленото произведение 119

4.3.5. Резултати от численото тестване на обобщения модел на метода на претеглената сума 121

Заключение – резюме на получените резултати 127

Приноси 130

Списък на публикациите по дисертационния труд 132

Декларация за оригиналност на резултатите 133

Библиография 134

УВОД

Използването на методи и модели от многокритериалното вземане на решения, е важно защото те предоставят систематичен подход за оценка на алтернативи. Друга важна тяхна характеристика е възможността им едновременно да отчитат множество критерии, които често са противоречиви. Такива са метода на претеглената сума и метода на претегленото произведение, върху които се съсредоточава настоящия дисертационен труд. Те се отличават с простота, прозрачност и широка приложимост в различни процеси, като бизнес, инженерни, социални, управленски и други. Независимо от широкото им приложение, класическите модели не успяват да отразят напълно реалните условия, в които са поставени лицата, вземащи решения (ЛВР). В практиката често се срещат ситуации, при които ЛВР са с различна компетентност, а за търсеното решение са поставени изисквания за удовлетворяване на ключови показатели от налични обективни характеристики. Тези особености определят необходимостта от разработването на модификации, които не само да повишат прецизността и обективността на взетото решение, но също така и да улеснят процеса по оценяване и по структуриране на задачата за вземане на решение.

За да се осигури практическа приложимост и достъпност на подобни модификации, повишаващи прецизността и обективността на взетото решение, е удачно те да бъдат имплементирани в уеб базирани приложения. Поради това е необходимо да се проектира подходяща софтуерна архитектура, на база на която да се реализира конкретното приложение, което да отговаря на изискванията за гъвкавост и мащабируемост. Чрез прилагане на утвърдени принципи и добри практики при проектирането на ясно дефинирани архитектури, може да се постигне минимизиране на риска от хаотични и трудно поддържани системи. Следователно, архитектурата е от решаващо значение за разработването и поддръжката на софтуерната система. Това е така, защото тя определя цялостната структура и взаимодействието между различните компоненти, позволявайки на разработчиците лесно и бързо да актуализират, модифицират или разширяват различни части на системата, без да нарушават други.

Отчитайки тясната връзка между моделите за подпомагане вземането на решения и софтуерните архитектури, става ясно, че е необходимо цялостно разбиране

както за процесите на вземане на решения, така и за софтуерните инструменти, имплементиращи тези модели.

Дисертационният труд е структуриран в увод, 4 глави, заключение – резюме на получените резултати, приноси, списък на публикациите, декларация за оригиналност на резултатите и библиография. Той съдържа 143 страници, 24 таблици, 53 фигури, от които 7 представят блокове на псевдокод и 143 литературни източника.

В Глава 1 е направен анализ на методите за вземане на решение и на софтуерните архитектури за реализиране на уеб приложения. Представени са различни концепции и класификации на методите за многокритериален анализ. По-специално внимание е отделено на метода на претеглената сума и метода на претегленото произведение. Направено е сравнение между двата метода и са разгледани някои техни модификации. Особено внимание е обърнато на класификациите на софтуерните системи за подпомагане вземането на решения (СПВР). Втората част от Глава 1 е с акцент върху софтуерните архитектури за реализиране на уеб приложения. Представен е анализ на концепциите за софтуерни архитектури, архитектурни стилове и шаблони. Въз основа на всичко това са определени изводи, на база на които е формулирана целта и задачите на настоящия дисертационен труд.

В Глава 2 са описани предложените модификации на метода на претеглената сума и метода на претегленото произведение, продиктувани от необходимостта за по-прецизно и относително обективно формиране на групово или в частност на индивидуално решение. Първата модификация отчита различията в компетентността на ЛВР спрямо различните критерии, като се използват теглови коефициенти за компетентност. Втората модификация въвежда използването на обективни данни, съчетавайки ги с предпочитанията на ЛВР. По този начин се редуцира субективността и се улеснява процеса по оценяване на алтернативите. Тази модификация се отнася единствено до метода на претеглената сума. По същество тя не променя математическия модел, а добавя формализация на генерирането на оценки, в алгоритъма за реализиране на метода. Третата модификация въвежда нов тип теглови коефициенти за алтернативите, базирани на ключови критерии, които са критични изисквания на конкретната задача. Чрез тези коефициенти, може да се постигне потискане или усилване на агрегираните резултати, което придава некомпенсаторни характеристики на моделите на претеглената сума и на претегленото произведение.

Едновременно с това се намалява броя на субективните оценки и се акцентира върху важността на ключовите изисквания в задачата. Модификацията е валидна за двата метода и променя, както алгоритъма за използването им, така и математическите модели. Всяка от предложените модификации е описана с идентична структура.

В Глава 3 са описани предложените и проектирани две различни софтуерни архитектури на СПВР. Проектирането на две архитектури за една и съща система е обосновано от необходимостта да се оцени и избере подходящия набор от технологии за разработка и експлоатационна среда, като се вземат предвид технологичните изисквания, очакваната натовареност и контекста на използване. Използвани са описаните в Глава 2 модификации, на база на които са формулирани комбиниран и обобщен математически модел. Тези модели определят ядрото на бизнес логиката за вземане на решение. Предложен е обобщен алгоритъм на математическите модели, като ключови стъпки от него са описани с псевдокод. Функционалността на СПВР е проектирана чрез адаптирани UML диаграми. Дефинирани са роли, функционални изисквания и случаи на употреба на системата. Описани са необходимите формуляри и последователността на взаимодействие между потребителя и системата.

В Глава 4 са представени проведените експерименти с предложените модификации върху алгоритъма за използване и моделите на метода на претеглената сума и метода на претегленото произведение. Описани са проведените експерименти използвайки: (1) модификация, отчитаща компетентността на ЛРВ в различни области на компетентност, при задача за избор на платформа за електронна търговия; (2) модификация, формализираща генериране на оценки за алтернативите, базирани на обективни данни и предпочитания на ЛВР, при задача за избор на монитор; (3) модификация, формализираща генериране на теглови коефициенти за алтернативите, базирани на обективни данни и предпочитания на ЛВР, при задача за избор на софтуерна рамка за уеб разработка; (4) обобщеният алгоритъм и математическите модели, комбиниращи модификациите; (5) разработен прототип на система, подпомагаща вземането на решения. Експериментите са проведени с три различни тестови задачи. За всяка от тях са формулирани отделни сценарии, реализирани с едни и същи обективни входни данни и параметри на задачата.

В заключението са обобщени получените резултати и са очертани някои насоки за бъдещи изследвания.

ГЛАВА 1.
АНАЛИЗ НА МЕТОДИТЕ ЗА ВЗЕМАНЕ НА РЕШЕНИЯ И СОФТУЕРНИТЕ АРХИТЕКТУРИ ЗА РЕАЛИЗИРАНЕ НА УЕБ ПРИЛОЖЕНИЯ

Методите за многокритериално вземане на решения (Multiple Criteria Decision Making, MCDM) предлагат математически обосновани подходи при задачи с множество критерии и алтернативи, като обхващат дискретни и непрекъснати проблеми. Различни проучвания показват широкото приложение на тези методи в различни области от практиката [Dang и др., 2025; Taherdoost и Madanchian, 2023; Sotoudeh-Anvari, 2022; Tan и др., 2021; Chourabi и др., 2018; Mardani и др., 2015]. Редица други изследвания със статистически характер, проведени върху различни индустрии показват, че методите на MCDM се използват най-вече при вземането на стратегически и тактически решения, докато на оперативно ниво се използват по-рядко [Khan и др., 2018; Fattoruso, 2022; Rodríguez-Carrillo и др., 2025]. Това се дължи на липсата на подходящи инструменти, които да позволяват ефективното използване на методите на MCDM на оперативно ниво. По-широкото им приложение в ежедневните задачи би изисквало техническа инфраструктура, която да осигури бърз, достъпен и прозрачен процес. Предоставянето на удобен и гъвкав инструмент, например уеб или мобилно приложение, може да спомогне за прилагането на тези научно обосновани методи.

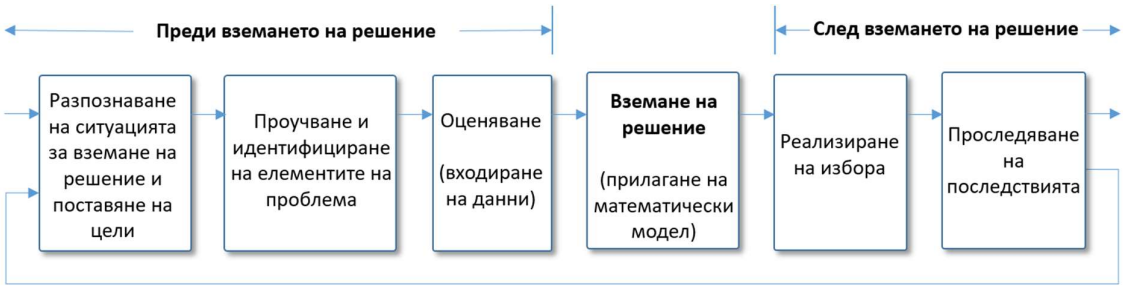
Разработката на уеб-базирани системи за подпомагане на вземането на решения отваря нови възможности за интегриране на утвърдени MCDM методи, предоставяйки интуитивен интерфейс, централизирано управление на данни и достъпност от всяка точка [Magabaleh и др., 2024]. При това, системата би могла да се адаптира към условията на индивидуално или групово вземане на решение. Такава система би позволила на лицата, вземащи решение да се фокусират върху същината на проблема. Останалите дейности, свързани с автоматизирането на рутинни операции като обработка на входни данни, нормализация и агрегиране, се делегират на самата софтуерна система. В допълнение тя може да улесни интерпретирането на резултатите, като по този начин повиши доверието в крайния избор.

1.1. Методи за подпомагане вземането на решения

През последните години се наблюдава нарастваща тенденция към интегриране на MCDM с анализ на данни, изкуствен интелект и машинно обучение за подобряване на процесите на вземане на решения [Amini и др., 2024]. Това отразява преминаването към подходи, базирани на повече данни, в различни области, включително икономика [Zayat и др., 2023], бизнес, здравеопазване [Chakraborty и др., 2023], управление и опазване на околната среда [Adu и др., 2025] и др. Наличието на разнообразие от методи за вземане на решение е предпоставка за направата на класификация на съществуващи концепции и методологии, както е показано в следващия раздел.

1.1.1. Концепции и класификации на методите за многокритериален анализ

Вземането на решения по множество критерии е клон на Изследване на операциите (Operations Research) [Peng и Shi, 2012]. Като научна област, MCDM изследва както рационалните, така и психологическите аспекти на този процес [Zavadskas и Turskis, 2011; Djartov и др., 2024]. Също така предполага необходимостта от съчетаване на обективни стойности със субективни предпочитания. Решенията могат да се вземат при различни условия в зависимост от различни аспекти, като например брой експерти, брой критерии и др. [Greco и др., 2016]. Вземането на решения включва множество процеси, като събиране на информация, моделиране на предпочитания, анализ на данни, и др. При това дейностите могат да се разделят на две фази – преди и след същинското вземане на решение както MacCrimmon отбелязва, илюстрирано с Фиг. 1.1.



Фиг. 1.1. Концептуален процес за вземане на решение [MacCrimmon, 1968].

Концептуалният процес разкрива широкия обхват на MCDM. По-късно Belton въвежда термина многокритериален анализ (Multiple-Criteria Decision Analysis, MCDA/MCA) [Belton и Stewart, 2002]. Двата термина често се използват

взаимозаменяемо, но MCDA стеснява рамката, фокусирайки се върху структурирането на проблема, оценяването и вземането на решение [Greco и др., 2025]. Всички MCDA проблеми обаче споделят няколко общи характеристики, посочени още от [Hwang и Yoon, 1981], които трябва да се вземат предвид както следва:

- **Множество цели или атрибути**: като под цели се разбират по-общите намерения на вземащия решение, изразяващи се чрез функция, която трябва да се оптимизира. Те изразяват какво трябва да се постигне, например минимизиране на разходите. Атрибутите са характеристиките, отразяващи до колко една алтернатива отговаря на поставен критерий, например цена;
- **Противоречиви критерии**: често срещано е в набора от критерии някои от тях да си противоречат;
- **Несъизмерими единици**: често критериите могат да включват различни мерни единици;
- **Проектиране или избор**: в случай, че съществува предварително дефиниран, краен набор от алтернативи, решението е да се избере най-добрата измежду тях. В обратния случай решението е да се проектира най-добра алтернатива.

Същите автори класифицират методите на MCDM в две направления в зависимост от задачите, които решават [Hwang и Yoon, 1981]:

(1) **Задачи за многокритериална оптимизация (Multiple Objective Decision Making, MODM)** – фокусирани върху пространства с непрекъснати решения и се решават посредством модели на математическо програмиране (линейно програмиране, целево програмиране и др.) или метаевристика [Ehrgott и др., 2025]. Работи се с цели, формулирани като функции за оптимизация;

(2) **Задачи за многокритериален анализ (Multiple Attribute Decision Making, MADM)** – дефинирани върху дискретни пространства за решения. Работи основно с атрибути, като наблюдавани и измерими характеристики на алтернативите.

За разлика от MODM, целта на MADM е да се вземе решение за предпочитание (т.е. сравнение, избор, приоритизиране, сортиране и/или класиране и т.н.) върху дискретно множество от алтернативи, характеризиращи се с множество, обикновено

противоречиви атрибути/критерии. От формална математическа гледна точка, процесът на MADM може да бъде формулиран както [Lopez и др., 2023]:

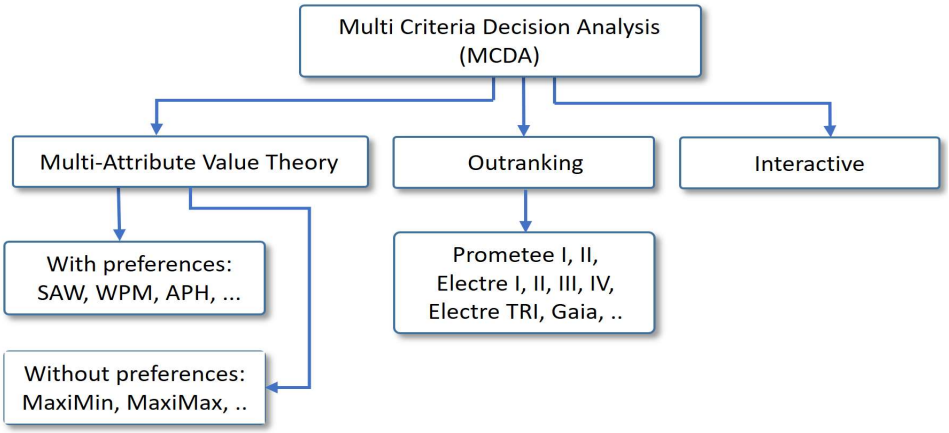
$$MADM \begin{cases} \text{избери} & \{A_1, A_2, \dots A_i \dots, A_I\} \\ \text{удоволетвори} & \{\, C_1, C_2, \dots C_j \dots, C_J \,\} \end{cases} \tag{1.1}$$

където A_i са I алтернативи на проблема, C_j са J атрибута (оттук нататък наричани критерии), които описват и характеризират алтернативите. Функцията „избери“ означава оптимизация (максимизиране или минимизиране) на стойност, която отразява полезността на дадена алтернатива и е дефинирана въз основа на множество критерии от лицата, вземащи решения (ЛВР) или други заинтересовани страни.

MADM не се ограничава до един метод, а представлява широка концепция, която обхваща различни методи със собствените им математически модели и алгоритми [Dean, 2022]. Многокритериалният анализ включва многобройни методи, някои от които са: Weighted Sum Model (WSM) [Triantaphyllou и Mann, 1989], Weighted Product Model (WPM) [Goswami и др., 2020], Analytical Hierarchy Process (AHP) [Saaty, 1987], Analytic Network Process (ANP) [Chen и др., 2019], VIseKriterijumska Optimizacija I Kompromisno Resenje (VIKOR) [Opricovic и Tzeng, 2004], Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) [Salih и др., 2019], Elimination Et Choix Traduisant la Realitè (ELECTRE) [Roy, 1991], Preference Ranking Organization Method for Enrichment Evaluation (PROMETHEE) [De Keyser и Peeters, 1996] и други.

Това разнообразие от методи, може да се систематизира по различни признаци, някои от които са описани от [Taherdoost и Madanchian, 2023; Cinelli и др., 2020]. В литературата се срещат трудове, които класифицират методите на MADM според признаци като например процедурата за вземане на решение [Ribeiro и др., 2011], теоретичната школа [Bączkiewicz и др., 2021], алгоритмичната и изчислителна сложност [Dean, 2022]. Съществено е разделението на методите на компенсаторни и некомпенсаторни [Hwang и Yoon, 1981]. Компенсаторните стратегии са широко приети, като рационална процедура на избор [Edwards, 1992; Keeney и Raiffa, 1993]. Алтернатива се предпочита, ако превъзхожда останалите в общата стойност на атрибутите, водейки до максимизиране на полезността като основен критерий за решение. При некомпенсаторните стратегии е важно алтернативата да превъзхожда останалите само по най-важните атрибути [Shiloh и др.,2001].

Също така с времето в MADM са се оформили и теоретични рамки като Multi Attribute Utility Theory (MAUT) [Fishburn, 1970] и Multi Attribute Value Theory (MAVT) [Ferretti, 2016; Keeney, 1996]. Основната концепция на MAUT, е че всяка алтернатива се оценява чрез функция на полезността, която комбинира множество атрибути и изборът е насочен към максимизиране на очакваната полезност. Прилага се, когато решенията трябва да отчитат несигурност или риск. В литературата MAVT често се разглежда като детерминистична версия на MAUT, където вместо функция на полезността се използва функция на стойността, измерваща резултата на алтернативите спрямо критериите [Dyer и др., 2005]. В съответствие с това, е представена една обобщена класификация, адаптирана от [Giove и др., 2009] на Фиг. 1.2.



Фиг. 1.2. Класификация на методите на MCDA.

В дисертационния труд предметът на изследването е съсредоточен върху подмножеството от методи, които могат да се определят като класически компенсаторни MCDA, а именно метода на претеглената сума (WSM) и метода на претегленото произведение (WPM).

1.1.2. Метод на претеглената сума и метод на претегленото произведение

Методът на претеглената сума и методът на претегленото произведение следват една и съща обща логика за оценяване на алтернативи въз основа на множество критерии, претеглени с коефициенти, отразяващи тяхната относителна важност. Както е отразено в имената им разликата между тях е в начина за агрегиране на резултатите.

В литературата, за метода на претеглената сума често се използват различни взаимозаменяеми имена, без да се прави ясно разграничение между тях – WSM, SAW (Simple Additive Weighting) и WLC (Weighted Linear Combination). Това произтича от

историческото им навлизане в различни дисциплини, което е наложило различни термини. Например [Malczewski, 1999] отбелязва, че WLC е по същество SAW/WSM и че в географските науки именно този термин е станал общоприет. При управлението на проекти методът най-често се назовава като WSM [Ahmad и Naimi, 2023; Malefaki и др., 2025], докато в инженерството преобладава наименованието SAW, както е изследвано в [Taherdoost, 2023]. С течение на времето в литературата започва да се прави разграничение между метод WSM или метод SAW [Zaidan и др., 2015]. Разликата между WSM и SAW се корени не в математическата формулировка, а в алгоритъма за използването им. В алгоритъма на SAW, присъства стъпката за нормализация на оценките на алтернативите спрямо критериите чрез математическа техника и ограничение за сумата на теглата да бъде равна на 1 [Taherdoost, 2023]. Докато при WSM нормализацията може да се изрази изцяло като ограничение върху субективно изразените входните данни за математическия модел, например [0, 1].

Една от най-ранните математически формулировки на първообраза на WSM се среща като сума от произведения, използвана за определяне на очакваната ефективност при избора на посока за развитие на дейността в областта на икономиката и статистиката. Тази формулировка е описана в труда на [Churchman и Ackoff, 1954]. Подходът бързо навлиза в практиката и е тестван в различни проблемни области [Eckenrode, 1965]. Концепцията за сума от претеглени произведения се въвежда в контекста на приоритизиране и оценяване при вземане на решения от [Fishburn, 1967]. Тази концепция получава алгоритмично описание и математически модел и се нарича процедура за сумирано претегляне (additive weighting procedure) от [MacCrimmon, 1968]. В последствие идеята се адаптира за целите на MCDM и в частност на MADM и в задачи за агрегиране на оценки, се извежда следната математическа формулировка за WSM [Triantaphyllou и Mann, 1989]:

$$\mathrm{A}_{i\,WSM} = \sum_{j=1}^{J} w_j a_{i,j} \qquad \forall i = 1, 2, \dots, I \tag{1.2}$$

където: $\mathrm{A}_{i\,WSM}$ изразява агрегираната оценка на i-*мата* алтернатива, изчислена чрез сума от произведенията от тегловия коефициент за важност на j-*тия* критерий – w_j и оценката дадена на i-*мата* алтернатива спрямо j-*тия* критерий – $a_{i,j}$.

В допълнение при WSM се предлага въвеждане на ограничения за стойностите на множителите чрез нормализация на входните стойности [Hwang и Yoon, 1981], като различни техники за избор на теглови коефициенти се предлагат от [Williams и Cai, 2025]. Също така, моделът намира приложение в условията на групово вземане на решения MCGDM (Multi-Criteria Group Decision Making) [Hwang и Lin, 1987, Kabak и Ervural, 2017]. В настоящият дисертационен труд, като ясна и изчерпателна дефиниция за класически модел на WSM, се приема следната математическа формулировка в общия случай на групово вземане на решение [Borissova, 2016]:

$$A_{i\,WSM}^{GDM} = \sum_{k=1}^{K} \sum_{j=1}^{J} w_j^k a_{i,j}^k \qquad \forall i = 1, 2, \dots, I \tag{1.3}$$

При следните ограничения, осигуряващи нормализирани стойности на входа:

$$\sum_{j=1}^{J} w_j^k = 1, \qquad \forall k = 1, 2, \dots, K \tag{1.4}$$

$$a_{i,j}^k, w_j^k \in [0, 1] \tag{1.5}$$

където: $A_{i\,WSM}^{GDM}$ изразява агрегираната оценка на i-*ата* алтернатива, изчислена чрез суми от произведенията от тегловия коефициент за важност на j-*тия* критерий според k-*тото* ЛВР – w_j^k и оценката дадена на i-*ата* алтернатива от k-*тото* ЛВР спрямо j-*тия* критерий – $a_{i,j}^k$. За нормализация се изисква оценката, дадена от ЛВР на всяка алтернатива спрямо конкретен критерий, както и коефициентите w_j^k, да се намират в диапазона от 0 до 1, включително. В допълнение, за да бъдат относителни, сумата на тегловите коефициенти за важност трябва да е равна на 1 (за всяко ЛВР).

Корените на WPM се проследяват до Bridgman от 1922 г. в контекста на анализ на размерностите и е адаптиран от Miller и Starr през 1969 г. към проблеми, свързани с вземане на решения [Triantaphyllou и Mann, 1989]. В някои статии ясно се отбелязва, че метода WPM се нарича още MEW (Multiplicative Exponent Weighting) [Zanakis и др., 1998; Wang и др., 2024]. В своята опростена форма, когато е освободен от ролята да осигурява безразмерност или нормализация на резултатите, методът WPM се представя със следната математическа формулировка в условията на индивидуално вземане на решение [Triantaphyllou, 2000]:

$$\mathrm{A}_{i\,WPM} = \prod_{j=1}^{J} \left(a_{i,j}\right)^{w_j} \qquad \forall i = 1, 2, \dots, I \tag{1.6}$$

Аналогично на методът WSM, в настоящия дисертационен труд, като ясна и изчерпателна дефиниция за класически модел на WPM, се приема следната математическа формулировка в общия случай на групово вземане на решение, заедно с ограниченията си:

$$A_{i\,WPM}^{GDM} = \prod_{k=1}^{K} \prod_{j=1}^{J} \left(a_{i,j}^{k}\right)^{w_j^k} \qquad \forall i = 1, 2, \dots, I \tag{1.7}$$

При същото ограничение за сумата на теглата (1.4) и още едно различно за диапазона на оценката от (1.5):

$$a_{i,j}^{k} = (0,1] \tag{1.8}$$

където: $A_{i\,WPM}^{GDM}$ изразява агрегираната оценка на i-*ата* алтернатива, изчислена чрез произведение от оценките, степенувани с коефициента за важност на критериите. Поради мултипликативния характер на метода, изискването всички оценки за алтернативите да са различни от 0, предотвратява автоматичното елиминиране на някоя от алтернативите [Tofallis, 2014].

1.1.3. Алгоритъм за вземане на решение чрез метода на претеглената сума и метода на претегленото произведение

Методите WSM и WPM споделят един и същ алгоритъм на цялостния процес по вземане на решение, като основната разлика между тях е в начина на агрегиране на оценките на алтернативите [Bozorg-Haddad и др., 2021]. В проучената литература, алгоритъмът често се представя с вариации по структура и детайлност, а мащабът на изложението варира в зависимост от източника [UK Government Analysis, 2024; Baker и др., 2001; Triantaphyllou, 2000; Mustakerov и Borissova, 2014; Singh и Malik, 2014]. За WSM и WPM, алгоритъма е стеснен до обхвата на MCDA [Belton и Pictet, 1997] като продължение на етапите на разпознаване на ситуацията за вземане на решение, проучване и идентификация на елементите ѝ [MacCrimmon, 1968]. В допълнение е ограничен до теоретичната рамка на MAUT и по-специално на MAVT [Jansen, 2011]. С оглед на това, алгоритъмът е адаптиран както е показано на Фиг. 1.3.

Фиг. 1.3. Концептуален алгоритъм за вземане на решение чрез методите WSM и WPM.

Стъпка 1 инициира събитие за многокритериално вземане на решение, което може да бъде в общия случай групово или в частност индивидуално. Тази стъпка структурира предварително дефинираните елементи на задачата, които включват:

- множеството на алтернативи;
- множеството на критерии;
- множеството на лица, вземащи решение;
- нормализационна техника;
- математически модел – избор между линеен (WSM) или мултипликативен (WPM);
- цел на анализа/природа на решението;

На **Стъпка 2** се извършва оценяване на алтернативите, което се изразява в задаване на стойност за алтернативата спрямо всеки от критериите. Стойностите могат да бъдат количествени за измерими показатели или качествени, които трябва да се трансформират в числови стойности [Baker и др., 2001; MacCrimmon, 1968]. Оценяването може да изразява субективно мнение или да се базира на обективни данни [Velasquez и Hester 2013]. Може да се приложат различни техники за събиране на данни за алтернативите и критериите от различни източници, например техники за машинно обучение, каквито са описани в литературния обзор по темата от [Liao и др., 2023]. В случай че оценките се определят субективно, стойностите им трябва да спазват ограниченията (1.5) при WSM или (1.7) при WPM.

Следва **Стъпка 3**, където се определя относителната важност на критериите чрез задаване на теглови коефициенти, спазвайки условието (1.4). Определянето на

тегловите коефициенти е критична стъпка тъй като различно зададените коефициенти могат напълно да променят крайния резултат от процеса по вземане на решение [Decancq и Lugo, 2009]. Подходи за обосновано определяне на теглата, базирано на обективни данни е предложен от [Tofallis, 2022; Pramanik и др., 2021].

Стъпка 4 се състои в съставяне на нормализирана матрица за вземане на решения. Въпреки че терминът съдържа думата „матрица“, в WSM и WPM не се извършват операции в смисъла на линейната алгебра, а се прилагат аритметични преобразувания като нормализация, претегляне и агрегиране върху елементите [Triantaphyllou и др., 1998]. Самият термин се е наложил исторически като удобен начин за реферирane на таблично представените данни. Таблицата позволява ясна и компактна систематизация на информацията и е изходна точка за прилагането на агрегиращия модел. Табличната структура на матрицата, в случая на групово вземане на решение може да има вида от Таблица 1.1 [Borissova, 2016].

Таблица 1.1. Матрица за вземане на решение при класическите методи WSM и WPM.

Критерии	**Тегла за важност на критериите**			**Оценки за алтернативите**						
				$\boldsymbol{A_1}$			…	$\boldsymbol{A_I}$		
	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	…	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$
C_1	w_1^1	…	w_1^K	$a_{1,1}^1$	…	$a_{1,1}^K$	…	$a_{I,1}^1$	…	$a_{I,1}^K$
…	…	w_j^k	…	…	$a_{i,j}^k$	…	…	…	$a_{i,j}^k$	…
C_J	w_J^1	…	w_J^K	$a_{1,J}^1$	…	$a_{1,J}^K$	…	$a_{I,J}^1$	…	$a_{I,J}^K$

Тъй като в общият случай критериите се измерват в различни мерни единици и мащаби, стойностите им трябва да бъдат нормализирани за да се получи коректно агрегиране на резултатите. За да се вземе предвид полезността на всеки критерии, повечето техники за нормализация предлагат две формулировки съобразно ориентацията на оптимизацията на критериите – максимизиращи (в полза) или минимизиращи (във вреда) [Aytekin, 2021]. При критериите в полза, по-високите стойности са предпочитани, докато при критериите във вреда, се предпочитат по-ниските стойности. Някои от най-често използваните нормализационни техники в контекста на MCDM са представени в Таблица 1.2 [Gardziejczyk и Zabicki, 2017; Jafaryeganeh и др., 2020; Aytekin, 2021; Vafaei и др., 2022].

Таблица 1.2. Нормализационни техники, зависими от ориентацията на полезността на критериите.

Нормализационна техника	Максимизиращ критерий (критерий в полза)	Минимизиращ критерий (критерий във вреда)
Линейна нормализация базирана на сума	$a_{i,j}^{benefit} = \frac{a_{i,j}}{\sum_{i=1}^{I} a_{i,j}}$ (1.9.а)	$a_{i,j}^{cost} = \frac{1/a_{i,j}}{\sum_{i=1}^{I} 1/a_{i,j}}$ (1.9.б)
Линейна нормализация базирана на максимум (Max)	$a_{i,j}^{benefit} = \frac{a_{i,j}}{a_{max}}$ (1.10.а)	$a_{i,j}^{cost} = 1 - \frac{a_{i,j}}{a_{max}}$ (1.10.б)
Линейна нормализация базирана на минимум (Min)	$a_{i,j}^{benefit} = 1 - \frac{a_{min}}{a_{i,j}}$ (1.11.а)	$a_{i,j}^{cost} = \frac{a_{min}}{a_{i,j}}$ (1.11.б)
Линейна нормализация, използваща Max и Min	$a_{i,j}^{benefit} = \frac{a_{i,j}}{a_{max}}$ (1.12.а)	$a_{i,j}^{cost} = \frac{a_{min}}{a_{i,j}}$ (1.12.б)
Линейна нормализация по Weitendorf (Max-Min)	$a_{i,j}^{benefit} = \frac{a_{i,j} - a_{min}}{a_{max} - a_{min}}$ (1.13.а)	$a_{i,j}^{cost} = \frac{a_{max} - a_{i,j}}{a_{max} - a_{min}}$ (1.13.б)
Векторна нормализация	$a_{i,j}^{benefit} = \frac{a_{i,j}}{\sqrt{\sum_{i=1}^{I} a_{i,j}^2}}$ (1.14.а)	$a_{i,j}^{cost} = 1 - \frac{a_{i,j}}{\sqrt{\sum_{i=1}^{I} a_{i,j}^2}}$ (1.14.б)
Логаритмична нормализация	$a_{i,j}^{benefit} = \frac{\ln(a_{i,j})}{\ln(\prod_{i=1}^{I} a_{i,j})}$ (1.15.а)	$a_{i,j}^{cost} = \frac{1 - \frac{\ln(a_{i,j})}{\ln(\prod_{i=1}^{I} a_{i,j})}}{I-1}$ (1.15.б)

На **Стъпка 5** се прилага избраният математически модел. Тук се изпълнява агрегиране на индивидуалните оценки на всяка алтернатива по отношение на всички критерии в една числова стойност, която изразява цялостната оценка за алтернативата. И при двата метода WSM и WPM, след прилагане на математическия модел за всяка алтернатива като резултат се получава вектор от скаларни стойности, както се изразява Engau в [Engau, 2010]. Този вектор служи като вход към последната **Стъпка 6** от алгоритъма. В литературата се подчертава, че природата на резултата от формираното решение може да приема различни форми в зависимост от зададената цел при дефинирането на задачата. Обикновено в задачите се търси една най-добра алтернатива, класиране или сортиране на всички алтернативи, а в някои случай се цели елиминиране на неподходящи алтернативи. Някои форми на резултата, още наричано

цел на анализа или природа на решението са представени в Таблица 1.3 [Hwang и Yoon, 1981; Triantaphyllou, 2000; Figueira и др., 2010; Talukder и др., 2017].

Таблица 1.3. Природа на формираното решение.

Форма на резултата	Формулировка
Избор на една най-добра алтернатива	$\mathrm{A}_{WSM/WPM} = \max[A_1 \dots A_i \dots A_I]$ (1.14)
Класиране/сортиране на всички алтернативи	$[A_1 > \cdots > A_i > \cdots > A_I]$ (1.15)
Елиминиране на неподходящи алтернативи (филтриране)	$[\mathrm{A}_{i\,WSM/WPM}] = [A_1 \dots A_i \dots A_I]$ където $A_i \geq \min A$ (1.16)

Приложението на WSM и WPM позволява не само избор и класиране на алтернативи, но и осигурява систематичен процес, който може да бъде адаптиран към различни контекст и задачи. Възможността за субективно или обективно определяне на оценките на алтернативите, разнообразните техники за нормализацията им и различните варианти за определяне на тегловите коефициенти, осигуряват гъвкавост и съвместимост с реални проблеми. Алгоритъмът може да се адаптира, както към индивидуално, така и за групово вземане на решение. Заради идентичните стъпки, които следват двата метода, паралелното им използване би позволило получаването на по-надеждни и валидирани резултати.

1.1.4. Сравнение на метода на претеглената сума и метода на претегленото произведение

Общовалидно за методите WSM и WPM е, че принадлежат към семейството на многокритериалното вземане на решения (MCDM). Попадат в подмножеството на многокритериалния анализ (MCDA/MCA), защото предлагат аналитична рамка, включваща функция за агрегиране и структуриран подход за оценка на алтернативите [Belton и Stewart, 2002]. От изчислителна гледна точка попадат и в подмножеството на многокритериалния анализ при дискретни алтернативи (MADM), тъй като задачите, които решават са с краен брой алтернативи, които се оценяват по множество атрибути/критерии [Hwang и Yoon, 1981]. Класифицират се като компенсаторни методи, тъй като високата оценка по един критерий може да компенсира ниска оценка по друг критерий [Mateo, 2012]. По своята същност представляват методи за оценяване на

алтернативи (Scoring methods) [Jafaryeganeh и др., 2020]. Тъй като използват тегла за изразяване на важността на критериите, се определят и като методи за претегляне (Weighting methods). Освен това те са методи за агрегиране на оценки (Aggregation methods), при което WSM следва адитивен (линеен) подход, а WPM – мултипликативен (нелинеен) подход.

Сравнение между WSM и WPM, което да показва техните характеристики, предимства и недостатъци, е представено в Таблица 1.4.

Таблица 1.4. Сравнение между WSM и WPM.

Характеристики	WSM	WPM
Агрегация	**Сума** от произведения на оценки на алтернативите и тегла на критериите. Линейна (адитивна).	**Произведение** от оценките, повдигнати на степен теглата на критериите. Нелинейна (мултипликативна).
Компенсация	Пълна компенсация. Висок резултат по един критерий може да компенсира нисък по друг.	Ограничена компенсация. Ниски стойности по един критерий трудно се компенсират.
Поведение към високи оценки	Подсилва резултата.	Подсилва леко.
Поведение към ниски оценки	Понижава резултата.	По-силно понижава резултата (силен наказателен ефект, особено при оценки близки до 0).
Нормализация на оценките	Необходима (за да бъдат безразмерни и сравними оценките).	По-рядко необходима. Не е задължителна [Tofallis, 2014].
Изисквания за данните	Могат да бъдат положителни или отрицателни.	Само положителни стойности, тъй като 0 елиминира алтернативата.
Чувствителност към интервала на измерване	Висока. Един критерий с голяма стойност може да доминира [Podvezko, 2011].	По-ниска, но много малки стойности силно намаляват резултата.
Интерпретация	Сума от претеглени оценки е по-лесна за интуитивна интерпретация.	Произведение от претеглени оценки е по-трудно за интуитивно възприемане.
Приложения	Оценка на проекти; Избор на доставчици; Финансов анализ. Най-често използван метод в практиката, подходящ за широк спектър от задачи [Aruldoss и др., 2013; Taherdoost, 2023].	Инженерство, Управление на риска. По-рядко използван, но полезен когато е важно да се избегне несъвместимост на единици и когато ниските оценки трябва да се наказват по-силно [Saleh и Hilal, 2024; Nasr и др., 2022].

Предимства	- Лесен за разбиране и прилагане; - Широко използван.	- Избягва прекомерна компенсация; - Подходящ за критерии, където всеки е критично важен; - Подходящ за разнородни измервателни единици.
Недостатъци	- Позволява прекомерна компенсация - Може да изкриви резултати при екстремни стойности	- Чувствителен към много ниски стойности (резултат клони към 0) - По-труден за разбиране и интерпретация.

Двата метода следват сходен процес на оценяване и избор на алтернативи. Съществената разлика между тях се състои в начина на агрегиране на оценките – при WSM/SAW/WLC се използва линейна комбинация (сумиране), докато при WPM/MEW се прилага мултипликативен подход (произведение с експоненти). За да бъдат съпоставими резултатите от двата метода, при прилагане върху една и съща задача, е необходимо резултатите от WPM да се трансформират чрез логаритмични преобразувания [Triantaphyllou и Sánchez 1997; Tofallis, 2014]:

$$\ln(A_{i\,WPM}) = \sum_{j=1}^{J} w_j \ln\left(a_{i,j}\right) \qquad \forall i = 1, 2, \dots, I \tag{1.17}$$

Това преобразуване улеснява интерпретацията, позволява по-лесно сравнение между алтернативите и дава възможност за прилагане на линейни аналитични техники, включително анализ на чувствителността.

1.1.5. Модификации над метода на претеглената сума и метода на претегленото произведение

Като едни от най-старите и фундаментални методи в MCDM, методите WSM и WPM претърпяват множество модификации и адаптации с други методи. Голяма част от тези изменения са насочени към: (1) **допълване или промяна в набора от атрибути, върху които се прилагат методите**; (2) **комбиниране с методи за работа с несигурност и неопределеност**; (3) **създаване на хибридни модели**.

Модификация, която адаптира методите WSM и WPM в условието на групово вземане на решение е предложена от [Borissova и др., 2018]. Въвеждат се тегла за всеки експерт, които отразяват различни нива на опит и познания по темата на проблема. Комбинирайки данни от множество експерти, подходът подобрява обективността и

приложимостта на методите в реални групови ситуации. Модификация, която засяга само метода WPM, наречена Ratio Product Model (RPM), е предложена в [Mohammadi и Rezaie, 2023]. Там авторите дават идеята оценките на алтернативите и теглата на критериите да се разглеждат като съотношения, а не като скаларни стойности. По този начин RPM избягва нуждата от нормализация както е при класическия WPM, която често влияе на резултатите. С този подход се гарантира, че крайното класиране остава стабилно, независимо от това как се мащабират данните. В друга модификация, валидирана само върху метода WSM от [Sorooshian и Parsia, 2019], се предлага информацията за оценките на алтернативите или теглата на критериите да идва от различни източници — например, експертни оценки, статистически данни или други външни ресурси. Методът въвежда допълнителни стъпки в алгоритъма на процеса, чрез които данните да се нормализират адекватно. Статията [Alanazi и др., 2013] представя модификация на WSM чрез въвеждането на динамични тегла, които се променят с времето. Този подход, наречен Dynamic WSM (DWSM), цели да отрази променящите се приоритети и условия в процеса на вземане на решения. В статията се представя математически модел за DWSM, който включва времеви фактори при определянето на теглата, и се обсъждат възможни приложения на този метод в различни области. В следния труд на [Arman и др., 2022] авторите представят Homogeneous SAW (H-SAW), където се налагат ограничения в степента на компромисите между критериите, което намалява компенсаторния характер на оригиналния метод. По този начин се намалява рискът една алтернатива с изключително добри стойности по няколко критерия да компенсира значително слабо представяне по други, което прави модела по-хомогенен. Методът е приложен в контекста на избор на доставчици, като е интегриран също с Fuzzy AHP.

Комбинацията между WSM и WPM в метода WASPAS (Weighted Aggregated Sum Product Assessment) има за цел да обедини предимствата на двата метода в един хибриден метод [Zavadskas и др., 2012]. Методът е определен като универсален и гъвкав благодарение на въведен параметър, който позволява настройване на степента на влияние на двата метода. Друг хибриден метод, комбиниращ WSM и WPM е WISP (Integrated Simple Weighted Sum Product) предложен от [Stanujkic и др., 2021]. Тук се използват опростени нормализационни процедури и се въвеждат четири мерки за полезност за по-прецизно класиране на алтернативите.

Интердисциплинарният подход на [Kaur и Kumar, 2013] въвежда метод, наречен Intuitionistic Fuzzy SAW (IFSAW), който разширява класическите SAW/WSM, като използва интуиционистки размити множества (IFS). Този подход позволява по-точно отразяване на несигурността и липсата на информация при оценките. Включва последователно конструиране на интуиционистки размита матрица за вземане на решение и прилагане на функцията за агрегиране от SAW/WSM. В статията си [Esangbedo и Che, 2016] предлагат модификация наречена Grey WSM, която разширява класическия метод, като заменя числените оценки със сиви числа (grey numbers). Целта е да се отчете несигурността и непълнотата на информацията в решения, засягащи икономическото развитие на Западна Африка. Методът позволява вземане на решения в среди с ограничени или неточни данни, като съчетава класическата формулировка на WSM с инструментите на Grey System Theory.

С течение на времето методът на претеглената сума и методът на претегленото произведение, са били развивани чрез различни модификации. Тяхната еволюция показва, че те не са забравени статични методи, а динамично се адаптират към нови практически потребности [Greco и др., 2025]. В резултат на това стават все по-гъвкави, широко приложими и подходящи да се интегрират в СПВР.

1.1.6. Системи за подпомагане вземането на решения

Системите за подпомагане на вземането на решения (Decision Support Systems, DSS) се разглеждат в литературата с двойствено значение – като концептуална рамка и като реална софтуерна система. Като концепция СПВР обхваща методи и модели за подпомагане на рационалния избор. Докато като софтуерна реализация, те се проявяват като системи, интегриращи данни, модели и аналитични инструменти [Turban и др., 2011]. В този контекст методите от MCDM са само един от възможните инструментите, интегрирани в СПВР. Въпреки това те имат ключова роля като математически обоснован начин за избор между алтернативи по множество критерии.

Еволюцията на софтуерно-базираните СПВР е представена като генеалогично дърво, като основните типове са категоризирани в хронологичен ред от [Arnott и Pervan, 2014]:

- Персонални СПВР (Personal Decision Support Systems, PDSS): обикновено са малки системи, предназначени за индивидуално използване при специфични задачи;
- Групови СПВР (Group Support Systems, GSS): включват комуникации, автономно гласуване и правят по-ефективна колективната дейност;
- СПВР за преговори (Negotiation Support Systems, NSS) осигуряват структуриран подход за обмен на предложения, оценка на компромиси и намиране на взаимноизгодни решения;
- Интелигентни СПВР (Intelligent Decision Support Systems, IDSS): интегрират изкуствен интелект – експертни системи, невронни мрежи, генетични алгоритми или размита логика;
- СПВР, базирани на управление на знания (Knowledge Management-Based DSS, KMDSS): за създаване, съхранение и използване на знания в организациите. Помагат за натрупване на експертен опит и добри практики;
- СПВР за бизнес разузнаване и анализ (Business intelligence, BI): подпомагат стратегически, тактически и оперативни решения, чрез трансформиране на данни в знание и конкурентно предимство.

Изследване фокусирано върху многокритериални системи за подпомагане на вземането на решения (MCDSS) е представено от [Razmak и Aouni, 2014]. Прегледът обхваща приложения на MCDSS в различни сектори, като стратегическо планиране, подбор на доставчици, планиране на търговски локации, финансови решения, здравеопазване и др. В тази връзка в статията на [Petković и др., 2025] се представя разработена софтуерна платформа наречена MCDM Solver, предназначена за подпомагане избора на биоматериали. Тя интегрира класически MCDM методи като TOPSIS, VIKOR и разширен WASPAS. Друга интересна СПВР за избор на MCDA методи, наречена MCDA-MSS се предлага от [Cinelli и др., 2022]. След като се опише проблема, системата разпознава най-подходящият метод от над 200 включени, базирайки се на множество характеристики. Разгледаните трудове демонстрират широкото приложение и практическата стойност от реализирането на методите на MCDM като софтуерен инструмент.

1.2. Софтуерни архитектури за реализиране на уеб приложения

В условията на бързо развиващата се информационна среда, уеб приложенията представляват основно средство за предоставяне на услуги, обработка на данни и взаимодействие между потребители и организации. Те са важен инструмент в контекста на СПВР, където се изисква интеграция на сложни алгоритми, обработка на големи обеми от данни и динамично взаимодействие с потребителите. Централна роля в разработването на такива системи има софтуерната архитектура, която определя как отделните компоненти на приложението си взаимодействат и как данните протичат между тях [Clements и др., 2010]. Изборът на подходящ архитектурен стил влияе пряко върху производителността, мащабируемостта и надеждността на уеб приложението. За да се направи обзор на архитектурите за уеб приложения, е необходимо да се разграничат няколко фундаментални понятия, свързани със структурата и организацията на софтуера.

1.2.1. Концепции на софтуерни архитектури, архитектурни стилове и шаблони

В литературата понятието за **софтуерна архитектура** се възприема като високо ниво на абстракция, което описва структурата на системата, нейните основни компоненти и начина, по който те си взаимодействат. Тя определя скелета на приложението и поставя рамката за вземане на решения относно практическата реализация [Bass и др., 2022]. Архитектурата е тясно свързана със **софтуерния дизайн**, като разликата е в нивото на абстракция. Докато архитектурата отговаря на въпроса „Как изглежда системата на високо ниво?“, дизайнът се концентрира върху въпроса „Как точно да се реализира всеки компонент?“. Софтуерният дизайн е по-детайлно ниво на описание, което разглежда реализацията на отделните модули и взаимодействието между тях [Cervantes и Kazman, 2024].

С течение на времето в софтуерното инженерство са се формирали различни **архитектурни стилове**, които задават основните принципи за проектиране на системите [Garlan и Shaw, 1994]. Сред най-разпространените категории стилове са монолитни, разпределени, събитийно-ориентирани и базирани на облачни технологии. Конкретни реализации в рамките на тези стилове се осъществяват чрез **архитектурни шаблони**. Тяхното възникване се разглежда като естествено

продължение на концепцията за шаблоните за дизайн на програмния код (Design Patterns), адаптирана от GoF [Gamma и др., 1995], но приложена на по-високо, архитектурно ниво. Архитектурните шаблони предоставят утвърдени решения за често срещани проблеми, като дефинират правила за организация и взаимодействие между компонентите [Tekinerdogan и Verdouw, 2020]. В практиката често се комбинират различни стилове и шаблони, което позволява извличане на предимствата им и компенсиране на слабостите. Този подход е известен в литературата като хибридни архитектури [Richards и Ford, 2020].

Всичко до тук описано, представя по-скоро идеализираната перспектива за това как софтуерните системи следва да бъдат проектирани. На практика обаче често се наблюдава противоположна ситуация, при която архитектурата се развива непреднамерено и придобива характеристиките на т.нар. Big Ball of Mud (BBoM), термин, въведен от [Foote и Yoder,1997; Richards и Ford, 2020]. Авторите идентифицират шест архитектурни шаблона, които илюстрират пропастта между теоретичните препоръки и реалността в практиката. Именно тази действителност подчертава нуждата от прилагане на утвърдени принципи и добри практики при проектирането на ясно дефинирани архитектури, които да минимизират риска от хаотични и трудно поддържани системи.

1.2.2. Софтуерни архитектурни стилове и шаблони

При проектирането на уеб приложения архитектурните стилове служат не само като „пътеводител“ за конкретните реализации, но и като основа за класифициране и сравнение на архитектурните решения. Въпреки това, обикновено се описват неформално, което може да доведе до различни интерпретации и погрешни заключения. За преодоляване на тези проблеми, следния доклад [Marmsoler, 2014] предлага развитие чрез създаване на строга теория на архитектурните стилове. В други изследвания се анализират стиловете, техните предимства и слабости, с цел въвеждане на по-голяма яснота и последователност [Kumar, 2014]. Също така има трудове, в които се набляга върху връзката между теорията и практиката, изследвайки приложимостта на различни стилове и шаблони за специфични задачи [Sharma и др., 2015].

Някои основни архитектурни стилове при проектирането на уеб приложения, заедно с често използвани шаблони, могат да се идентифицират, базирайки се на

различни източници като [Garlan и Shaw, 1994; Buschman и др., 1996; Clements и др., 2010; Richards и Ford, 2020; Bass и др., 2022; и други]. Например, на Фиг. 1.4 е представен един от възможните варианти за класифициране на софтуерни архитектурни стилове и шаблони за реализиране на уеб приложения.

Фиг. 1.4. Софтуерни архитектурни стилове и шаблони за реализиране на уеб приложения.

Важно е да се отбележи, че изложението на Фиг. 1.4 е условно и в практиката няма ясно разграничение между архитектурен стил и архитектурен шаблон, както и това къде свършва архитектурния стил и къде започва архитектурата [Perry и Wolf, 1992].

Многослоен архитектурен стил (Layered Architecture Style), организира архитектурата в поредица от подредени слоеве, като всеки слой има ясно дефинирани отговорности и комуникира единствено със съседните си слоеве. Основната цел е да се постигне разделяне на отговорностите, така че логиката на системата да бъде структурирана и по-лесно управляема. Сред предимствата на този стил са

преизползваемост, улеснена поддръжка, разширяемост. Недостатъците включват възможна загуба на производителност поради множеството междинни стъпки и тенденция към увеличена сложност при прекалено много слоеве [Clements и др., 2010]. Някои практически аспекти на налагането на този стил в софтуерното инженерство са категоризирани от [Tu, 2023]. Примери за N-tier шаблони, разделящи отговорностите по различен начин спрямо характера на проектираната система, са [Kouraklis, 2016]:

- MVC (Model–View–Controller) – класическото разделение на бизнес логика, потребителски интерфейс и база данни;
- MVP (Model–View–Presenter) – еволюция на MVC с фокус върху визуализацията на данните;
- MVVM (Model–View–ViewModel) – комбинира принципите от MVC и MVP, с фокус върху интерактивно сложно взаимодействие с потребителя.

Архитектурният стил „Поточна линия“ (Pipeline Architecture Style), организира обработката на данни като последователност от свързани етапи, при които изходът на един етап служи като вход за следващия. Всеки етап изпълнява определена трансформация на данните, а комуникацията обикновено се осъществява чрез буфери или потоци. Също като многослойния стил и тук се използва йерархична топология, но фокусът е върху обработката и трансформирането на непрекъснати данни. Предимствата са ясно разделение на задачите, лесно добавяне/премахване на етапи, висока степен на преизползваемост. За недостатъци могат да се приемат липсата на гъвкавост при динамични зависимости и потенциални проблеми с производителността, ако етапите са неравномерно натоварени [Kumar, 2014]. Типични приложения включват компилатори, конвертори на файлове, обработка на големи обеми от данни в реално време. Най-често прилаганите шаблони, са [Behera и Vardhan, 2025]:

- Pipe-and-Filter – данните се обработват последователно и поточно от независими компоненти, в ролята на филтри, свързани чрез канали (pipes);
- Batch Sequential – данните се обработват последователно на партиди, което е подходящо за пакетна обработка на големи обеми от данни;
- ETL (Extract-Transform-Load) – включва извличане, трансформация и зареждане на данни от различни източници с фокус производителност.

Архитектурният стил, базиран на услуги (Service-Based Architecture Style, SBA Style), се използва за изграждане на разпределени системи чрез самостоятелни или слабо свързани услуги. Основната концепция е приложението да се раздели на отделни услуги, всяка от които реализира конкретна бизнес функция и може да бъде разработвана, внедрявана и мащабирана независимо от останалите [Buschman и др., 1996]. Често се използва за приложения, при които различни модули трябва да бъдат независими и същевременно интегрирани. Основните предимства, които има са модулност, независимо внедряване и повторна употреба. Недостатък може да бъде повишената сложност на интеграцията и управлението на зависимости между модулите. Въпреки че е спорно това дали следва да се разглеждат като архитектурни шаблони или като самостоятелни стилове, в практиката се срещат автори [Richards, 2015], които класифицират следните решения като шаблони в рамките на SBA стила:

- **Архитектура ориентирана към услуги** (Service Oriented Architecture, SOA), често се внедрява в големи организации със сложни бизнес процеси. Компонентите, които предоставят услуга и тези, които я използват са независими, често разработвани на различни езици и платформи. Обикновено инфраструктурата е централизирана и се използват комплексни комуникационни стандарти като SOAP и WSDL [Clements и др., 2010];
- **Архитектура от микроуслуги** (Microservices Architecture), често се разглеждат като естествена еволюция на SOA, но за разлика от нея, има децентрализирана инфраструктура. Топологията често е мрежова, като всяка услуга може да взаимодейства с други чрез API (Application programming interface). Използват се по-опростени комуникационни протоколи, като HTTP/REST. Концепцията за декомпозиране се разширява, като бизнес функциите се разделят на по-малки съставни части, представляващи микроуслуги [Fowler и Lewis, 2014].

Безсървърният архитектурен стил (Serverless Architecture Style), се използва за изграждане на разпределени системи и освобождава разработчиците от управлението на сървърната инфраструктура като го прехвърля към доставчика на облачни услуги. Основната идея е, че разработчиците се фокусират върху бизнес логиката и приложните функции, без да се налага да конфигурират, администрират или поддържат сървъри. При безсървърният стил често голяма част от бизнес логиката се реализира като

приложение от една страница (Single-Page Application, SPA) и се изпълнява в браузъра на клиента [Jiang и др., 2020]. Предимствата на този стил включват автоматично мащабиране спрямо натоварването, цена заплащана единствено за реално използвани ресурси, ускорена разработка. Недостатъци са по-трудното отстраняване на грешки и въвеждане на подобрения в производителността, заради липсата на пълен контрол над средата. Някои архитектурни шаблони, често използвани в този стил са:

- SPA – управлява взаимодействието с потребителя като променя съдържанието на страницата динамично чрез JavaScript. Комуникира със сървърите чрез API без да зарежда нови страници;
- FaaS (Function-as-a-Service) – задълбочава концепцията за декомпозиране на бизнес логиката, до степен малки независими функции, които се задействат от събития. Дава решения за ефимерна еластичност, при пикови натоварвания [Wawrzoniak и др., 2024];
- BaaS (Backend-as-a-Service) – изнася голяма част от функционалностите на бизнес логиката като готови услуги от външен доставчик.

Архитектурният стил задвижван от събития (Event-Driven Architecture Style, EDA), въвежда концепцията за събития като основен механизъм за комуникация и координация между отделните компоненти. При него системата реагира на настъпили събития като промяна на състоянието, потребителско взаимодействие или външен процес, обработвайки ги асинхронно чрез специализирани компоненти (event handlers). Стилът осигурява висока гъвкавост, мащабируемост и слаба свързаност между компонентите. Като недостатъци могат да се посочат сложно проектиране и трудно откриване и отстраняване на грешки. Особено ефективен е при системи, работещи в реално време и IoT и безсървърни приложения [Richards и Ford, 2020]. Често използвани архитектурни шаблони са [Gamma и др., 1995]:

- Publish–Subscribe – компонентите, създаващи събития ги публикуват, а консуматорите се абонират за тези събития, от които се интересуват;
- Event Bus – представлява централизирана инфраструктура, която приема събития от множество създатели и ги разпространява към консуматорите;
- Observer Pattern – компонентите, създаващи събития известяват всички консуматори за настъпило събитие, които решават дали да го обработят.

Архитектурният стил Microkernel разделя системата на базова функционалност (микро ядро), която е минималната необходима за функциониране и разширения, които се включват като допълнителни модули (plug-in). Този стил е характерен за системи, при които е необходимо лесно надграждане и адаптиране към променящи се изисквания. Той е често използван при операционни системи, уеб браузъри, платформи и софтуерни продукти, където базовата система трябва да остане стабилна, а разширенията независими. Основни предимства са гъвкавост и лесно надграждане, а предизвикателствата включват сложност при управлението на взаимодействието между ядрото и модулите [Buschman и др., 1996].

В заключение може да се отбележи, че не съществува универсално най-добър архитектурен стил или шаблон. Изборът винаги зависи от конкретните функционални и нефункционални изисквания на разработваното приложение. В практиката рядко се реализират архитектури в чист вид. По-често се разчита на комбинации от различни стилове и шаблони, които се смесват или допълват взаимно, за изграждането на уеб приложения.

1.2.3. Характеристики за сравнение на софтуерни архитектури

При проектирането на уеб приложения изборът измежду голямото разнообразие от взаимно преплитащи се архитектурни стилове и шаблони, представлява сам по себе си сложна многокритериална задача. Изследване в тази посока е направено от [Galster и др., 2010]. Авторите предлагат метода SYSAS, който систематизира процеса на подбор, съпоставяйки характеристиките на архитектурните елементи с изискванията на крайния потребител. В тази връзка е необходимо да бъдат разгледани ключови характеристики, които могат да служат като критерии за сравнение между софтуерните архитектури и за дефиниране на нефункционални изисквания. Автори като [Richards и Ford, 2020; Bass и др., 2022] описват в детайли и категоризират множество характеристики, като едни от най-универсалните са:

Мащабируемост (scalability) – способността на системата да поддържа нарастващ брой потребители или обем от данни чрез добавяне на ресурси. Разпределените архитектури, като SBA и безсървърните предлагат висока степен на мащабируемост, докато монолитните имат слабо представяне по тази характеристика.

Производителност (performance) – способността на системата да реагира бързо и ефективно. Монолитите архитектури обикновено имат висока производителност за малки приложения. В SBA има забавяния заради комуникацията между услугите. При безсървърните производителността зависи от доставчика на облачните услуги. Мащабируемостта заедно с производителността са критични аспекти на съвременните уеб приложения, гарантиращи, че приложенията могат да се справят с нарастващото потребителско търсене, като същевременно поддържат оптимална скорост и ефективност [Shethiya, 2025].

Разгръщане (deployment) – отнася се до това, колко лесно може системата да се внедрява и управлява в реална среда. При монолитните архитектури изпълнението на бизнес логиката е концентрирана на едно място. При SBA логиката е разпределена между услуги на различни сървъри, а при безсървърните архитектури логиката се реализира като облачни функции.

Надеждност и устойчивост (reliability, resilience) – способността на системата да продължи работа и функционира коректно при неизправности. Разпределените архитектури правят частични откази и позволяват локализиране и изолиране на грешки, докато монолитните стигат до пълни сривове.

Сигурност (security) – способността на системата за защита на данни и услуги от злонамерени действия. Монолитните архитектури предполагат по-лесно централизирано управление на сигурността, докато разпределените изискват сложни механизми за удостоверяване и сигурна комуникация между услугите. Хофман предоставя информация относно някои уязвимости и начини за ограничаване на рисковете за няколко технологии за изграждане на уеб приложения, като GraphQL, облачни внедрявания, мрежи за доставяне на съдържание (CDN) и рендериране от страна на сървъра (SSR) [Hoffman, 2024].

Обработка на събитията (event handling) – отнася се до начина, по който системата реагира на настъпили събития. Тя може да бъде синхронна, при която изпълнението изчаква завършването на текущата обработка, преди да премине към следващото събитие, това е типично за Pipeline архитектурите. При асинхронната обработка отделните събития се обработват независимо и паралелно, без необходимост от изчакване, което е типично за EDA архитектурите. Някои автори предлагат подход за проектиране, основаващ се на разделяне на отговорността между

части от програмния код за изпълнение на синхронни и асинхронни операции [Mokrushin и Ilinykh, 2024].

Разработка и поддръжка (development, support) – отнася се до лекотата, с която системата може да бъде разработвана, разширявана, модифицирана и поддържана през целия си жизнен цикъл. Монолитните системи са по-лесни за първоначално разработване, но трудни за поддръжка и еволюция при растеж. SBA улесняват разделянето на работата между екипи и непрекъснатото внедряване. Безсървърните предлагат най-бързо внедряване, но с ограничения за дългосрочна поддръжка.

Софтуерните характеристиките често влизат в конфликт, например между сигурност и производителност. Това налага архитектът да взема решения чрез компромиси и приоритизиране на конкретните изисквания на проектираната система.

1.3. Изводи

В резултат на извършения обзор, съсредоточен върху метода на претеглената сума и метода на претегленото произведение, се установява, че те са ясно дефинирани, утвърдени и попадат в разнообразни класификации. Широкото им приложение и множеството модификации показват, че тяхното изследване и развитие е перспективно. Разнообразните подходи за субективно и обективно оценяване, нормализация и определяне на тегла, осигуряват гъвкавост и адаптивност към различни задачи, както за индивидуални, така и за групови решения. Въпреки че двата метода са много популярни, прилагането им към оперативни решения в ежедневието изостава. Това е предпоставка за реализирането на WSM и WPM в софтуерен инструмент, като модифицирани по начин, улесняващ ЛВР да ги използва, би имало значима практическа стойност. Тъй като тези модели предполагат интензивни изчислителни операции, често в реално време, както и необходимост от прозрачност и сътрудничество между множество заинтересовани лица, изборът на подходяща софтуерна архитектура се превръща в ключов фактор. Архитектурното решение трябва да гарантира не само коректна обработка и агрегиране на данните, но и висока производителност, надеждност и удобство за потребителите.

1.4. Цел и задачи

Цел на дисертационният труд е да се предложат модели и софтуерни архитектури на системи за подпомагане вземането на решения. За реализиране на целта е необходимо да се изпълнят следните задачи:

1) да се направи анализ на метода на претеглената сума и метода на претегленото произведение, както и на основните софтуерни стилове и шаблони за разработка на уеб приложения;
2) да се предложи модификация на алгоритъма и моделите на WSM и WPM, отчитаща различните области на компетентност на ЛВР;
3) да се предложи модификация на алгоритъма на WSM за формализиране на генерирането на оценки за алтернативите;
4) да се предложи модификация на алгоритъма и моделите на WSM и WPM за формализиране на генерирането на коефициенти, даващи предимство в общото представяне на алтернативите;
5) да се предложи обобщен алгоритъм и комбинирани математически модели, съчетаващи зададените като цел модификации;
6) да се проектират различни софтуерни архитектури, които да удовлетворяват противоречиви изисквания към системата за подпомагане вземането на решения;
7) да се разработят прототипи на СПВР в съответствие с проектираните архитектури, включително модул за интерпретация на резултатите;
8) да се определи практическата приложимост на предложените модификации, реализирани в софтуерни архитектури на уеб приложения чрез провеждане на числени тестове на задачи от практиката.

ГЛАВА 2.
МОДИФИКАЦИИ НА МЕТОДА НА ПРЕТЕГЛЕНАТА СУМА И МЕТОДА НА ПРЕТЕГЛЕНОТО ПРОИЗВЕДЕНИЕ

Едни от най-популярните представители на методите за оценяване в многокритериалното вземане на решения са метода на претеглената сума и метода на претегленото произведение. През годините редица изследвания потвърждават, че те остават сред най-често и широко прилаганите методи в практиката [Hwang и Yoon, 1981; Triantaphyllou и Mann, 1989; Bozorg-Haddad и др., 2021; Taherdoost, 2023]. Поради това дисертационният труд се съсредоточава върху тях и в настоящия раздел са описани предложените три модификации на двата метода. Предложените нови модификации са продиктувани от необходимостта за по-прецизно и относително обективно формиране на групово или в частност на индивидуално решение. Целта е предложените модификациите да се имплементират в система за подпомагане вземането на решения, която да е в състояние да отчита спецификите на конкретните задачи, целящи определяне на най-предпочитаната алтернатива.

Първата предложена модификация има за цел да отчете различията в компетентността на ЛВР по различните критерии. Това се реализира като се въвеждат теглови коефициенти за компетентност. В допълнение, за да се подпомогне администрирането на процеса и да се облекчат изчисленията се предлага механизъм за групиране на критериите. Модификацията е приложима както към WSM, така и към WPM. Промените спрямо класическите методи засягат както алгоритъма за прилагане, така и математическата формулировка на модела.

Втората предложена модификация цели да редуцира субективността и да улесни ЛВР при оценяването на алтернативите. Модификацията се отнася единствено до метода на претеглената сума. Тя не променя математическия модел, а добавя формализация на генерирането на оценки, в алгоритъма на метода. В традиционния подход се изисква ЛВР да определя числови оценки по всеки критерий, което е не само субективно, но и трудоемко. Предложеното решение включва типизиране на критерии

и генериране на оценки чрез съчетаване на обективни данни и предпочитания на ЛВР. Това позволява използване на реални обективни данни в модели, създадени преди дигиталната ера, и адаптирането им към съвременна среда с широк достъп до информация.

Третата предложена модификация въвежда нов тип теглови коефициенти за алтернативите, базирани на ключови критерии, които са критични за изпълнението на конкретната задача. Чрез тези коефициенти, може да се постигне потискане или усилване на агрегираните резултати на алтернативите, което придава некомпенсаторни характеристики на моделите WSM и WPM. В същото време се намалява броя на субективните оценки и се подчертава важността на ключовите изисквания в задачата. Модификацията е валидна за двата метода и променя, както алгоритъма за използването им, така и математическите модели.

Всяка от предложените модификации е описана с идентична структура. Процесът по вземане на решение е илюстриран и описан алгоритмично, отразявайки нововъведението. Формулиран е математическият модел на всяка от предложените модификации. Структурирани са изменените матрици на алтернативите за общия случай на групово вземане на решение и са представени новите матрици на декомпозираните до опции критерии.

2.1. Модификация на метода на претеглената сума и метода на претегленото произведение, отчитаща различните области на компетентност на ЛВР

При структурирането на задачи за многокритериално групово вземане на решения, се предполага наличието на изчерпателен набор от атрибути/критерии, чрез който се гарантира, че са взети предвид възможно най-много аспекти на разглеждания проблем [Keeney и Gregory, 2005]. Това условие изисква използване на критерии от различни предметни области на познание. Следователно, при формирането на група от компетентни лица се очаква, те да са доказани експерти в своята област на познание, но не винаги те са с еднаква степен на компетентност.

Един от съществуващите модели за формиране на групово решение, който отчита различията в компетентността на експертите, е описан в [Borissova и др., 2018].

Недостатък на този модел е факта, че компетентността на всички експерти, участващи при формирането на груповото решение, се третира като глобален фактор за цялата задача. Този недостатък може да се преодолее, като компетенциите на членове на групата се вземат предвид поотделно за отделните критерии. За постигане на това, е целесъобразно да се използват коефициенти, които да отразяват степента на компетентност на всяко едно ЛВР. Тези коефициенти трябва да са съобразени със съответния опит и област на познания. По този начин, нововъведените коефициенти се отнасят към всеки критерий и всички участници в процеса на групово вземане на решение [Dimitrova и др., 2023; Dimitrova и др., 2024, б].

При задачи с голям брой критерии, процесът по вземане на решение би се усложнил, тъй като трябва да се определят множество коефициенти. С цел да се намали времето и да се подпомогне администрирането на процеса, критериите могат да се групират по области на компетентност. Този процес на групиране води до същия ефект на разпределяне от математическа гледна точка. В последствие ЛВР получава един и същ коефициент за компетентност спрямо всички критерии, които съставят групата. Описаното до тук води до необходимостта от алгоритъм за реализиране на този процес.

В следващият раздел е описан предложения алгоритъм за отчитане компетентността на ЛВР при формиране на групово решение, отчитайки компетентността им в различни области на познанието.

2.1.1. Алгоритъм за реализиране на модификацията, отчитаща различията в областите на компетентност на ЛВР

Въз основа на концептуалния алгоритъм, представен на Фиг. 1.3, се предлага модифициран алгоритъм, който реализира модификацията за отчитане степента на компетентност на ЛВР в различни области. Той е илюстриран чрез схематичната диаграма на Фиг. 2.1.

Фиг. 2.1. Алгоритъм за реализиране на модификацията, отчитаща компетентността на ЛВР в различни области на компетентност.

Тъй като модификацията се прави в рамката на MCDA, процесът на проучване и формулиране на задачата не е включен в алгоритъма. **Първата стъпка** се свежда единствено до описание на параметрите на задачата за групово вземане на решение, включващи алтернативи, критерии и лица, вземащи решение. Начинът по който изброените параметри са определени, е извън обхвата на предложения алгоритъм. Същото важи и за бизнес анализа и мотивацията, довели до инициране на процеса по вземане на решения, както и за определянето на целта на вземането на решение и интерпретацията на крайния резултат.

Решението дали да се групират критериите или не, се взема в зависимост от изискванията на задачата и най-вече от броя критерии. Към **Стъпка 2** се преминава, когато не е необходимо или възможно да се съставят групи. В този случай тегловите коефициенти за компетентност се разпределят пропорционално между ЛВР за отделните критерии.

В случай, че критериите са много на брой и е възможно тяхното групиране, се преминава към **Стъпка 3**. Един добър вариант е групирането на критериите да се осъществи на база на „речник на компетенциите“ [Spencer и Spencer, 1993], определящ необходимия експертен опит.

На **Стъпка 4** се определят коефициентите за компетентност на групата от ЛВР, като тази итерация се изпълнява за всяка от вече дефинираните групи критерии. Самото определяне на коефициентите също не е предмет на настоящия труд. За тези цели например, може да се изпълни методично измерване на компетенциите [McClelland, 1973] в комбинация с личностно тестване [Briggs-Myers и Briggs, 1995].

Следващите стъпки от алгоритъма следват стъпките на класическите методи.

На **Стъпка 5** ЛВР дава оценка на алтернативите спрямо критериите.

На **Стъпка 6** ЛВР определят тегловите коефициенти за относителна важност на критериите, спрямо личната си преценка.

Следва **Стъпка 7**, където се съставя матрицата, съдържаща оценки на алтернативите, тегла за важност на критериите, както и нововъведените тегла за компетентност на ЛВР.

На **Стъпка 8**, се прилага избраният за задачата математически модел (WSM или WPM), модифициран така, че да отразява различията в компетентностите на ЛВР.

На **последната стъпка** се агрегират резултати за алтернативите, въз основа на които се формира груповото решение.

2.1.2. Математическа формулировка на модифицираните методи, отчитащи различията в областите на компетентност на ЛВР

При математическите модели на класическите методи WSM (1.3) – (1.5) и WPM (1.7) – (1.8) не са предвидени коефициенти, отчитащи компетентността на лицата, вземащи решения. В модификация, като следната [Borissova и др., 2018] са въведени коефициенти, изразени като вектор $\lambda^k = \{\lambda^1, \lambda^2, \dots, \lambda^k, \dots\}$. Елементите на този вектор

са скаларни стойности, които отчитат нивото на компетентност на всяко ЛВР, като не се прави разлика в компетентностите за отделните критерии.

В настоящия дисертационен труд се предлага модификация, чрез която глобалния за задачата коефициент за компетентност на ЛВР, от скаларна стойност да се превърне във вектор. По този начин, вместо да се разглежда индивидуалният принос на едно ЛВР към оценките по критериите с еднаква важност, коефициентът, който изразява неговата компетентност λ^k, ще се разглежда като вектор $\lambda_j^k = \{\lambda_1^k, \lambda_2^k, \dots, \lambda_j^k, \dots\}$. Елементите на новия вектор съдържат коефициенти, чиито стойности съответстват на нивото на компетентност на ЛВР спрямо всеки един критерий.

При тази постановка, векторът $\{\lambda^k\}$ се превръща в матрицата (двумерен масив) $\{\lambda_j^k\}$. Следователно, математическата формулировка на предложения модифициран модел на метод на претеглената сума придобива следния вид:

$$A_{i\,ModifiedWSM}^{GDM} = \sum_{k=1}^{K} \sum_{j=1}^{J} \lambda_j^k w_j^k a_{i,j}^k \tag{2.1}$$

При ограничения:

$$\sum_{j=1}^{J} w_j^k = 1, \qquad \forall k = 1, 2, \dots, K \tag{2.2}$$

$$\sum_{k=1}^{K} \lambda_j^k = 1, \qquad \forall j = 1, 2, \dots, J \tag{2.3}$$

$$a_{i,j}^k, w_j^k, \lambda_j^k \in [0, 1] \tag{2.4}$$

където:

- $A_{i\,ModifiedWSM}^{GDM}$ изразява обобщената оценка на i-*мата* алтернатива, изчислена по така модифицирания метод на претеглената сума;
- индекс i обозначава текущата алтернатива от общо I на брой и $i = 1, 2, \dots I$;
- индекс k обозначава *k-тия* ЛВР от общо K на брой и $k = 1, 2, \dots K$;
- индекс j обозначава *j-тия* критерий от общо J на брой и $j = 1, 2, \dots J$;
- λ_j^k е теглови коефициент, който отразява нивото на компетентност на k-*мото* лице, вземащо решение спрямо j-*тия* критерий;

— w_j^k е теглови коефициент, който отразява относителната важност на *j-тия* критерий според *k-тото* лице, вземащо решение;

— $a_{i,j}^k$ е оценката дадена на *i-тата* алтернатива от *k-тото* лице, вземащо решение спрямо *j-тия* критерий.

Следвайки същата логика, математическата формулировка на предложения модифициран модел на метода на претегленото произведение придобива следния вид:

$$A_{i\,ModifiedWPM}^{GDM} = \prod_{k=1}^{K}\prod_{j=1}^{J}\left(a_{i,j}^k\right)^{w_j^k \lambda_j^k} \tag{2.5}$$

При същите ограничения (2.2), (2.3) и още едно относно оценката:

$$a_{i,j}^k = (0,1] \tag{2.6}$$

където:

— $A_{i\,ModifiedWPM}^{GDM}$ изразява обобщената оценка на i-тата алтернатива, калкулирана по така модифицирания метод на претегленото произведение.

Въз основа на Таблица 1.1, данните за прилагане на така формулираните модификации на двата математическите модела (на WSM и на WPM) са структурирани в следната матрица за вземане на решение:

Таблица 2. 1. Матрица за вземане на решение при модификацията, отчитаща различията в областите на компетентност на ЛВР.

Критерии	**Тегла за компетентност на ЛВР**			**Тегла за важност на критериите**			**Оценки за алтернативите**						
							$\boldsymbol{A_1}$			…	$\boldsymbol{A_I}$		
	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	…	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$
C_1	λ_1^1	…	λ_1^K	w_1^1	…	w_1^K	$a_{1,1}^1$	…	$a_{1,1}^K$	…	$a_{I,1}^1$	…	$a_{I,1}^K$
…	…	λ_j^k	…	…	w_j^k	…	…	$a_{i,j}^k$	…	…	…	$a_{i,j}^k$	…
C_J	λ_J^1	…	λ_J^K	w_J^1	…	w_J^K	$a_{1,J}^1$	…	$a_{1,J}^K$	…	$a_{I,J}^1$	…	$a_{I,J}^K$

Прилагайки **Стъпка 3** и **Стъпка 4** от горе-описания алгоритъм (Фиг. 2.1) математическите изрази (2.1) – (2.4), се трансформират, така че да се отчетат коефициентите за компетентност на ЛВР по групи. Модификацията на модела на претеглената сума се представя по следния начин:

$$A_{i\,ModifiedWSM}^{GDM} = \sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{m_l=1}^{M_l} \lambda_l^k w_{m_l}^k a_{i,m_l}^k \tag{2.7}$$

При ограничения:

$$\sum_{l=1}^{L}\sum_{m_l=1}^{M_l} w_{m_l}^k = 1, \qquad \forall k = 1, 2, \dots, K \tag{2.8}$$

$$\sum_{k=1}^{K} \lambda_l^k = 1, \qquad \forall l = 1, 2, \dots, L \tag{2.9}$$

$$a_{i,m_l}^k, w_{m_l}^k, \lambda_l^k \in [0, 1] \tag{2.10}$$

където:

- индекс l обозначава l-*мата* група критерии от общо L на брой и $l = 1, 2, \dots L$;
- чрез m_l е изразен m-*тия* критерий от l-тата група от критерии, съдържаща M_l на брой критерии и $m_l = 1, 2, \dots M_l$;
- $w_{m_l}^k$ е коефициент, който отразява относителната важност на m_l-*тия* критерий според k-*тото* лице, вземащо решение;
- λ_l^k е теглови коефициент, който отразява нивото на компетентност на k-*тото* лице, вземащо решение спрямо l-*мата* група от критерии;
- a_{i,m_l}^k е оценката дадена на i-*мата* алтернатива от k-*тото* лице, вземащо решение спрямо m_l-*тия* критерий.

В съответствие с така описаното по-горе, математическият модел на модифицирания метод на претегленото произведение придобива следния вид:

$$A_{i\,ModifiedWPM}^{GDM} = \prod_{k=1}^{K}\prod_{l=1}^{L}\prod_{m_l=1}^{M_l} (a_{i,m_l}^k)^{w_{m_l}^k \lambda_l^k} \tag{2.11}$$

При същите ограничения (2.8), (2.9) и допълнителното:

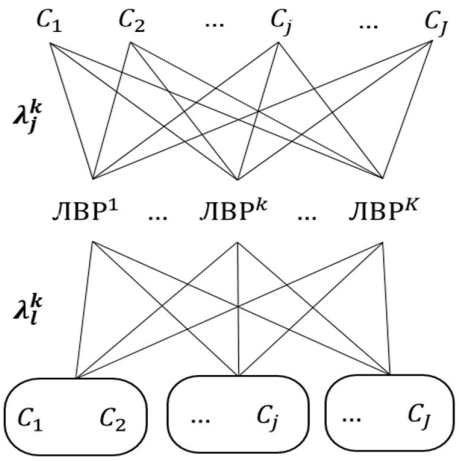
$$a_{i,m_l}^k = (0,1] \tag{2.12}$$

След въвеждането на групи от критерии матрицата за вземане на решение от Таблица 2.1 се модифицира до вида, показан в Таблица 2.2:

Таблица 2. 2. Матрица за вземане на решение при модификацията, отчитаща различията в областите на компетентност на ЛВР при групиране на критериите.

Групи критерии	**Критерии**	**Тегла за компетентност на ЛВР**			**Тегла за важност на критериите**			**Оценки за алтернативи**						
								$\boldsymbol{A_1}$			…	$\boldsymbol{A_I}$		
		ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	…	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$
G_1	C_1	λ_1^1	…	λ_1^K	$w_{1_1}^1$	…	$w_{1_1}^K$	$a_{1,1_1}^1$	…	$a_{1,1_1}^K$	…	$a_{I,1_1}^1$	…	$a_{I,1_1}^K$
	…				…	…	…	…	…	…	…	…	…	…
	C_{m_l}				…	…	…	…	…	…	…	…	…	…
…	…	…	λ_l^k	…	…	$w_{m_l}^k$	…	…	a_{i,m_l}^k	…	…	…	a_{i,m_l}^k	…
G_L	C_{m_L}	λ_L^1	…	λ_L^K	…	…	…	…	…	…	…	…	…	…
	…				…	…	…	…	…	…	…	…	…	…
	C_{M_L}				$w_{M_L}^1$	…	$w_{M_L}^K$	a_{1,M_L}^1	…	a_{1,M_L}^K	…	a_{I,M_L}^1	…	a_{I,M_L}^K

На Фиг. 2.2. са представени взаимовръзките между ЛВР, критериите, групите критерии и определените теглови коефициенти за компетентност.



Фиг. 2.2. Взаимовръзки между ЛВР, критериите, групите и теглата за компетентност.

2.2. Модификация на метода на претеглената сума, формализираща генериране на оценки за алтернативите

Предложената модификация на WSM, формализираща генериране на оценки за алтернативите се базира на обективни данни и предпочитания на ЛВР, чрез декомпозиране на критерии до опции.

При традиционните оценъчни методи (методи за агрегиране), базирани на теорията на многоатрибутната полезност, каквито са изследваните, се изисква ЛВР да определи оценки на алтернативите спрямо критериите [Yudhistira и др., 2024]. Тези оценки се приемат за субективни, тъй като зависят изцяло от личната преценка на ЛВР.

В ерата на изобилие от данни, методите за тяхното събиране, структуриране и анализ се развиват бурно [Chen и др., 2014; Wu и др., 2014]. В резултат на този напредък възниква възможността данните да бъдат включени в класическите модели за вземане на решения, първоначално разработвани в условия на по-ограничени източници на обективна информация. Това е мотивацията да се предложи модификация, изследваща вариантите, субективната оценка да бъде заменена от обективни данни съчетани с предпочитанието на ЛВР. От тук следва, че предложения метод е валиден само в случая на задачи с изцяло обективни по своята същност критерии. Освен че се намалява субективния фактор, процесът по оценяване на алтернативите е значително улеснен. От ЛВР се изисква единствено изразяване на предпочитания, а не задаване на оценки. От друга гледна точка, този подход позволява алтернативите да останат в тайна от ЛВР, ако това е изискване на задачата [Dimitrova и др., 2024, б].

Възможни са два начина за генериране на оценка. Когато данните за алтернативата са количествени, тяхната нормализирана стойност се приема за оценка. Ролята на ЛВР е да изрази мнението си за това дали стойностите са в полза или вреда според него. В съответствие с това, при прилагане на нормализационната техника се изпълнява максимизация или минимизация на полезността по атрибута/критерия. При другият вид генерирани оценки, обективните данни отразяват наличието или липсата на качество, характеристика, свойство или най-общо казано опции на алтернативата. Затова опциите се представят с двоична стойност. На свой ред ЛВР изразява предпочитанията си по този списък от възможни опции за множеството на

алтернативите. Отново следва обективните данни да се съчетаят с предпочитанието на ЛВР в генерирана оценка.

Критичен момент, извън обхвата на изследването е съобразяването на избраната нормализационна техника със спецификите на всеки един критерии поотделно. Както и процеса по събиране и анализ на данните.

2.2.1. Алгоритъм за реализиране на модификацията, формализираща генериране на оценки за алтернативите

Предложената модификация може да се използва за решаване на практически задачи, следвайки стъпките на алгоритъма представен чрез схемата на Фиг. 2.3.

Фиг. 2.3. Алгоритъм за реализиране на модификацията, формализираща генериране на оценки за алтернативите.

Алгоритъмът **започва** с определяне на параметрите на задачата за вземане на решение, а именно определяне на алтернативите, критериите и лицата, вземащи решение. Идентично на алгоритъма описан в точка 2.1. начините, по които изброените параметри са определени, бизнес анализа и мотивацията, довели до инициране на процеса по вземане на решения са извън обхвата на предложения алгоритъм.

На **Стъпка 2** е необходимо да се съберат и анализират обективните данни за алтернативите във връзка с критериите за оценка. Анализа е необходим, за да могат данните да добият вид, който е подходящ за целите и обхвата на конкретната задача за вземане на решение. Добър вариант е да се използват методите за събиране и анализ на данни, разработени от дисциплините статистика и наука за данните.

На **Стъпка 3**, се определя типът на критерия. В предложената модификация има възможност да се разграничат два типа критерии в зависимост от начина, по който са представени данните за оценка на алтернативите:

- ✓ Един критерий е от тип **числено представен,** ако за него са налични количествени данни за всички оценявани алтернативи. Също така трябва да изпълнява условието, при нормализация да може да се изразява предпочитание за максимизиране или минимизиране на стойностите.
- ✓ Критерият е от тип **матрично представен,** ако данните за алтернативите отразяват наличието на качество или не може да се определи дали предпочитаната количествена стойност е в полза или вреда.

В случай че критерият е числено представен, се преминава към **Стъпка 4,** където ЛВР определя дали се търси максимизация или минимизация на полезността на критерия/атрибута за оценка при нормализация на данните.

Следва **Стъпка 5**, при която се избира подходяща нормализационна техника и се прилага върху обективните количествени данни, като се взема предвид предпочитанието за полезност. Резултатът е генерирана оценка на алтернативата спрямо съответния числено представен критерий.

В случай че критерият е матрично представен, се продължава към **Стъпка 6**, където критерият се декомпозира до опции, извлечени от всички алтернативи. Важно условие е всяка от опциите да присъства в поне една алтернатива. По този начин, списъкът на опциите не нараства излишно. След като опциите са определени, се съставя

двоична матрица, която съпоставя наличието на опция в дадена алтернатива със стойност 1, а отсъствието със стойност 0.

На следващата **Стъпка 7**, ЛВР изразява предпочитанията си за това, колко е важно всяка от опциите да бъде налична в алтернативите. Предпочитанието се изразява, като се зададат оценъчни точки от предварително определена скала.

Оценка за алтернатива спрямо матрично представен критерии се генерира на **Стъпка 8**. Необходимо е стълбовете от двоичната матрица, представяща опциите да бъдат умножени по вектора ред, съдържащ оценъчните точки, разпределени от ЛВР и резултатът се разделя на максималния възможен такъв.

Двете разклонения на алгоритъма се събират в **Стъпка 9**, от където се продължава идентично с алгоритъма на Фиг. 2.1., ЛВР определят относителни теглови коефициенти за важност между всички критерии.

На **Стъпка 10** се попълва матрицата за вземане на решение и се прилага модифицираният математически модел на WSM.

Процесът **завършва**, като се агрегират оценките на отделните алтернативи, за да се получи крайният резултат.

2.2.2. Математическа формулировка на модифицирания метод, формализиращ генериране на оценки за алтернативите

При класическият модел на метода на претеглената сума (1.3) – (1.5) не се уточнява начинът, по който се определят оценките за алтернативите спрямо критериите $a_{i,j}^{k}$ зададени от ЛВР. Новост в настоящата дисертация, е предложението за модификация на WSM с възможност за интегриране на два подхода за генериране на оценки за алтернативите спрямо критериите. Генерираните оценки се базират на обективни данни, като същевременно отчитат индивидуалното предпочитание/мнение на ЛВР.

В общия случай, при груповото вземане на решение, математическият модел на модифицирания метод на претеглената сума придобива следния вид:

$$A_{i\ ModifiedWSM}^{GDM} = \sum_{k=1}^{K} \sum_{j=1}^{J} w_j^k \beta_{i,j}^k \tag{2.13}$$

При ограничения:

$$\sum_{j=1}^{J} w_j^k = 1, \qquad \forall k = 1, 2, \dots, K \tag{2.14}$$

$$\beta_{i,j}^k, w_j^k \in [0, 1] \tag{2.15}$$

където:

- $A_{i\,ModifiedWSM}^{GDM}$ изразява обобщената оценка на i-*ата* алтернатива, изчислена по така модифицирания метод на претеглената сума;
- индекс i обозначава i-*ата* алтернатива от общо I на брой и $i = 1, 2, \dots I$;
- индекс k обозначава k-*тия* ЛВР от общо K на брой и $k = 1, 2, \dots K$;
- индекс j обозначава j-*тия* критерий от общо J на брой и $j = 1, 2, \dots J$;
- w_j^k е теглови коефициент, който отразява относителната важност на j-*тия* критерий според k-*тото* лице, вземащо решение;
- $\beta_{i,j}^k$ е генерираната оценка на i-*ата* алтернатива, отчитаща предпочитанията на k-*тото* лице, вземащо решение спрямо j-*ти* критерий.

В зависимост от типа на критерия числено или матрично представен, разграничаваме два типа генерирани оценки:

$$\beta_{i,j}^k = \begin{cases} (b_{i,j}^k)^{norm} & b \in [0, 1] \\ c_{i,j}^k & c \in [0, 1] \end{cases} \tag{2.16}$$

където:

- $(b_{i,j}^k)^{norm}$е генерираната оценка по критериите от тип числено представен;
- $c_{i,j}^k$ е генерираната оценка по критериите от тип матрично представен.

В зависимост от предпочитанието на ЛВР, оценката $(b_{i,j}^k)^{norm}$ може да приеме една от двете нормализирани стойности след прилагане на подходяща нормализационна техника за задачата (вж. (1.9.а) – (1.15.б)):

$$(b_{i,j}^k)^{norm} = \begin{cases} maximize\ b_{i,j}^k \\ minimize\ b_{i,j}^k \end{cases} \tag{2.17}$$

Оценката на алтернатива спрямо матрично представен критерии, $c_{i,j}^k$ се изчислява по следната формула:

$$c_{i,j}^k = \frac{\sum_{r_j=1}^{R_j} s_{r_j}^k v_{i,r_j}}{R_j S} \tag{2.18}$$

При ограничения:

$$v_{i,r_j} \in 0|1 \tag{2.19}$$

$$\sum_{r_j=1}^{R_j} v_{i,r_j} \neq 0, \qquad \forall i = 1, 2, \dots, I \tag{2.20}$$

$$\sum_{i=1}^{I} v_{i,r_j} \neq 0, \qquad \forall r_j = 1, 2, \dots, R_j \tag{2.21}$$

където:

- броячът r_j обозначава текущата опция от общия брой R_j $(r_j = 1, 2, \dots R_j)$, съдържащи се в j-*тия* матрично представен критерий;
- $s_{r_j}^k$ е векторът с оценки, които отразяват важността на r_j-*ата* опция според k-*тото* лице, вземащо решение, спрямо предварително определена скала за оценяване с максимална оценка S;
- v_{i,r_j} е стойността на i-*ата* алтернатива спрямо, r_j-*ата* опция на j-*тия* матрично представен критерий.

Таблицата на декомпозираните до опции критерии, на база на която се изчисляват генерирани оценки на алтернативите, в общия случай на задача за групово вземане на решения е представена в Таблица 2.3.

Таблица 2.3. Таблица на критериите, декомпозирани до опции, по които се генерират оценки на алтернативите.

Матрично представени, декомпозирани критерии	**Опции**	**Двоични стойности на опциите спрямо алтернативите**			**Предпочитания на ЛВР относно опциите**		
		$\boldsymbol{A_1}$	…	$\boldsymbol{A_I}$	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$
C_1	O_1	$v_{1,1}$	…	$v_{I,1}$	s_1^1	…	s_1^K
	…	…	…	…	…	…	…
	O_{R_j}	…	…	…	…	…	…
…	…	…	v_{i,r_j}	…	…	$s_{r_j}^k$	…
C_J	O_1	…	…	…	…	…	…
	…	…	…	…	…	…	…
	O_{R_j}	v_{1,R_j}	…	v_{I,R_j}	$s_{R_j}^1$	…	$s_{R_j}^K$

Данните за прилагане на математическият модел на WSM могат да се структурират в следната матрица за вземане на решение, както е показано в Таблица 2.4.

Таблица 2.4. Матрица за вземане на решение при модификация, въвеждаща генериране на оценки на алтернативите, базирани на обективни данни и предпочитания на ЛВР.

Критерии		**Полезност на критериите при нормализация**			**Тегла за важност на критериите**			**Генерирани оценки за алтернативите**						
								$\boldsymbol{A_1}$			...	$\boldsymbol{A_I}$		
Тип	**Критерий**	ЛВР$^{\mathbf{1}}$	...	ЛВР$^{\boldsymbol{K}}$	ЛВР$^{\mathbf{1}}$	...	ЛВР$^{\boldsymbol{K}}$	ЛВР$^{\mathbf{1}}$	...	ЛВР$^{\boldsymbol{K}}$	...	ЛВР$^{\mathbf{1}}$	...	ЛВР$^{\boldsymbol{K}}$
числено представени	C_1	max/min	...	max/min	w_1^1	...	w_1^K	$b_{1,1}^1$	...	$b_{1,1}^K$	...	$b_{I,1}^1$	...	$b_{I,1}^K$
	...	...	...	...	...	...	...	...	$b_{i,j}^k$	...	...	...	$b_{i,j}^k$	...
	C_j	max/min	...	max/min	...	w_j^k	...	$b_{1,j}^1$	...	$b_{1,j}^K$	...	$b_{I,j}^1$	...	$b_{I,j}^K$
матрично представени	C_j	-			...	...	...	$c_{1,j}^1$	...	$c_{1,j}^K$	...	$c_{I,j}^1$	...	$c_{I,j}^K$
	...				...	...	...	...	$c_{i,j}^k$	...	...	...	$c_{i,j}^k$	...
	C_J				w_J^1	...	w_J^K	$c_{1,J}^1$	...	$c_{1,J}^K$	...	$c_{I,J}^1$	...	$c_{I,J}^K$

Тъй като модификацията допуска генерираната оценка на алтернативата да има стойност 0, това прави модификацията неприложима за метода на претегленото произведение.

2.3. Модификация на метода на претеглената сума и метода на претегленото произведение, формализираща генериране на коефициенти, даващи предимство в общото представяне на алтернативите

Предложената модификация на методите WSM и WPM формализира генерирането на теглови коефициенти, даващи предимство в общото представяне на алтернативите въз основа на обективни данни и предпочитанията на ЛВР, чрез декомпозиране на критерии до опции.

При реализирането на тази модификация се определят ключови критерии, спрямо които се генерират нововъведените коефициенти, прилагани към агрегираните оценки на алтернативите. За ключови критерии се определят онези, които са особено значими за конкретната задача и представляват задължителни изисквания или

изразено предпочитание на инициатора на задачата за вземане на решение. Това допълва класическите методи WSM и WPM, които са обект на изследването. От математическа гледна точка, начинът за генериране на оценка по матрично представен критерий (от предходната модификация) може да се модифицира за генериране на теглови коефициент за предимство в общото представяне на алтернативите [Borissova и др., 2021; Dimitrova и др., 2024, б]. Защо се предлага този коефициент да се генерира, а не да се определи субективно? Отговорът е – за да се даде възможност на предпочитанията да формират решението, а не обратното [Sharot и др., 2010].

С нарастването на броя на критериите, процесът на определяне на относителната им важност чрез теглови коефициенти става по-сложен. Това важи и за определянето на субективните оценки, които ЛВР трябва да даде за представянето на всяка алтернатива. Поради нарастващата сложност на реалните задачи и необходимостта от ускоряване на процеса по вземане на решение, предложения нов подход води до следните положителни ефекти: (1) Автоматично изключване или намаляване на оценката на алтернативи, които не отговарят на предварително зададени условия по ключови критерии. В същото време, при необходимост, влиянието на такива критерии може да бъде усилено при агрегиране на крайните резултати. (2) Определянето на относителната важност на критериите става по-лесно, тъй като се намалява броят им. (3) Намалява се броят на субективните оценки, които ЛВР трябва да определи за алтернативите спрямо критериите.

2.3.1. Алгоритъм за реализиране на модификацията, формализираща генериране на коефициенти, даващи предимство в общото представяне на алтернативите

Алгоритъмът, който описва предложената модификация, е представен чрез схемата илюстрация на Фиг. 2.4.

Фиг. 2.4. Алгоритъм за реализиране на модификацията, формализираща генериране на коефициенти за алтернативите.

Алгоритъмът **започва** с дефиниране на задачата за вземане на решение, което е идентично с първата стъпка от алгоритмите описани в настоящата глава т. 2.1.1 и т. 2.2.1.

Стъпка 2 е свързана с извличането на данни и анализа им. На база този анализ се опредял дали за задачата има критерии, които са ключови. Също така се събира информация за опциите.

На **Стъпка 3** следва да се разграничат критериите в зависимост от тяхното предназначение:

- ✓ за оценяване на алтернативи – каквито са критериите в класическите методи WSM и WPM;

- ✓ за предимство на алтернативи – в настоящата модификация още наричани ключови критерии, отразяващи характеристики/опции на алтернативите, които са особено значими с изисквания или ограничения за конкретната задача.

В случай че критерия се използва за оценяване на алтернативите, се преминава към **Стъпка 4**, където ЛВР определя своята субективна оценка за представянето на алтернативите спрямо зададените критерии.

На **Стъпка 5** ЛВР определя относителната важност на критериите, като ги отнасят един спрямо друг, на база на субективни предпочитания.

В случай че критерият се използва за предимство на алтернативите, той бива изключен от множеството на критериите, върху които се изпълняват стъпки 4 и 5. Тогава се преминава към **Стъпка 6**, където критерият се декомпозира до опции извлечени от всички алтернативи. Съставя се матрица, чиито стойности могат да бъдат 1 или 0. Наличието на опция в дадена алтернатива се изразява със стойност 1, а отсъствието със стойност 0. Важно е да се отбележи, че не се допуска наличието на опция, която не е налична в нито една от алтернативите, и обратното – алтернатива, която не съдържа нито една от опциите. Тези ограничения са наложителни, за да не се изключи предпочитание на ЛВР от модела или да се лиши алтернатива от предимство.

На **Стъпка 7,** ЛВР изразява предпочитанията си за това, колко е важно всяка от опциите да бъде налична в алтернативите, като определя коефициенти за относителната й важност.

Следва **Стъпка 8**, където за всеки критерий за предимство на алтернативи, се генерират коефициенти, сумата от които да е равна на 1.

Пътят към финализирането преминава през **Стъпка 9**, където се съставя матрицата за вземане на решение.

На **Стъпка 10** се прави избор и се прилага един от двата модифицирани модела на WSM или на WPM.

В **резултат** на решаването на задачата се получава и формираното групово решение.

2.3.2. Математическа формулировка на модифицираните методи, формализиращи генериране на коефициенти, даващи предимство в общото представяне на алтернативите

Тръгвайки от класическите методи на WSM (1.3) – (1.5) и WPM (1.7) – (1.8), може да се отбележи, че липсва механизъм, чрез който да се придава допълнителна тежест на алтернативите според някои критерии. Това може да бъде критично за някои задачи, при които има ограничения или изисквания. Друга слабост на класическите методи, е че сложността на определянето на тегловите коефициенти за важност на критериите w_j^k, нараства с увеличаването на броя им J. Същото важи и за броя на оценките на алтернативите спрямо критериите $a_{i,j}^k$.

Настоящата модификация изследва възможността спрямо някои от критериите да се пропусне задаване на оценки за алтернативите. Вместо това се дава възможност за изразяване на предпочитание, като се въвежда нов коефициент, който да отразява относителната важност между опциите, които алтернативите предлагат. По този начин в съчетание с данните и нововъведения коефициент, е възможно по-прецизно да се изрази цялостното представяне на алтернативата. Като страничен и положителен ефект се явява намаляването на множеството на критериите, за които ЛВР трябва да определят коефициенти за относителна им важност и респективно намаляване на оценяването на алтернативите заради намаления брой критерии.

В общия случай, при груповото вземане на решение, математическият модел на модифицираният метод на претеглената сума, съгласно горните разсъждения, придобива следния вид:

$$A_{i\,ModifiedWSM}^{GDM} = \sum_{k=1}^{K}\left(\psi_i^k \sum_{j=1}^{J} w_j^k\, a_{i,j}^k\right) \tag{2.22}$$

$$\psi_i^k = \prod_{n=1}^{N}\sum_{r_n=1}^{R_n} p_{r_n}^k v_{i,r_n} \tag{2.23}$$

При ограничения:

$$\sum_{r_n=1}^{R_n} p_{r_n}^k = 1, \qquad \forall n = 1,2,\dots,N;\ \ \forall k = 1,2,\dots,K \tag{2.24}$$

$$v_{i,r_n} \in 0|1 \tag{2.25}$$

$$\sum_{r_n=1}^{R_n} v_{i,r_n} \neq 0, \qquad \forall i = 1, 2, \dots, I \tag{2.26}$$

$$\sum_{i=1}^{I} v_{i,r_n} \neq 0, \qquad \forall r_n = 1, 2, \dots, R_n \tag{2.27}$$

$$\sum_{j=1}^{J} w_j^k = 1, \qquad \forall k = 1, 2, \dots, K \tag{2.28}$$

$$a_{i,j}^k, w_j^k \in [0, 1] \tag{2.29}$$

където:

- $A_{i\,ModifiedWSM}^{GDM}$ изразява обобщената оценка на i-*ата* алтернатива, изчислена по така модифицирания метод на претеглената сума;
- индекс i обозначава текущата алтернатива от общо I на брой и $i = 1, 2, \dots I$;
- индекс k обозначава k-*тия* ЛВР от общо K на брой и $k = 1, 2, \dots K$;
- индекс n обозначава n-*тия* критерий от общо N на брой, предназначени за придаване на предимство ($n = 1, 2, \dots N$);
- ψ_i^k е генерирания теглови коефициент за i-*ата* алтернатива на k -*тото* лице, вземащо решение, който съчетава относителната важност на r_n-*ата* опция на n-*тия* критерий и v_{i,r_n} стойността при i-*ата* алтернатива;
- индекс r_n обозначава r-*ата* опция на n-*тия* критерий, съдържащ R_n на брой опции и $r_n = 1, 2, \dots R_n$;
- $p_{r_n}^k$ е теглови коефициент, който отразява важността на r_n-*ата* опция според k -*тото* лице, вземащо решение;
- v_{i,r_n} е стойността на i-*ата* алтернатива спрямо, r_n-*ата* опция на n-тия матрично представен критерий, предназначен за предимство на алтернативите;
- индекс j обозначава j-*тия* критерий, предназначен за оценяване от общо J на брой и $j = 1, 2, \dots J$;
- w_j^k е теглови коефициент, който отразява относителната важност на j-*тия* критерий за оценяване според k -*тото* лице, вземащо решение;
- $a_{i,j}^k$ е оценката дадена на i-*ата* алтернатива от k-*тото* лице, вземащо решение спрямо j-*тия* критерий.

Математическият модел на модифицирания метод на претегленото произведение придобива следния вид:

$$A_{i\,ModifiedWPM}^{GDM} = \prod_{k=1}^{K} \psi_i^k \prod_{j=1}^{J} \left(a_{i,j}^k\right)^{w_j^k} \tag{2.30}$$

При същото калкулиране на коефициента за предимство на алтернативите (2.23) и ограничения (2.24) – (2.28) и още едно относно оценката.

$$a_{i,j}^k = (0,1] \tag{2.31}$$

където:

— $A_{i\,ModifiedWPM}^{GDM}$ изразява обобщената оценка на i-тата алтернатива, калкулирана по така модифицирания метод на претегленото произведение.

В общия случай на задача за групово вземане на решение, таблицата на критериите, декомпозирани до опции, на база на която се изчисляват допълнителните коефициенти за предимство на алтернативите, изглежда по следния начин (вж. Таблица 2.5):

Таблица 2.5. Таблица на критериите, декомпозирани до опции, с цел изчисляване на коефициенти за предимство на алтернативите.

Матрично представени, декомпозирани критерии	**Опции**	**Двоични стойности на опциите спрямо алтернативите**			**Тегла за важност към декомпозираните опциите**		
		$\boldsymbol{A_1}$	…	$\boldsymbol{A_I}$	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$
C_1	O_1	$v_{1,1}$	…	$v_{I,1}$	p_1^1	…	p_1^K
	…	…	…	…	…	…	…
	O_{R_n}	…	…	…	…	…	…
…	…	…	v_{i,r_n}	…	…	$p_{r_n}^k$	…
C_N	O_1	…	…	…	…	…	…
	…	…	…	…	…	…	…
	O_{R_n}	v_{1,R_n}	…	v_{I,R_n}	$p_{R_n}^1$	…	$p_{R_n}^K$

След така изчислените допълнителни коефициенти на база Таблица 2.5 и израз (2.22), може да се състави матрицата за вземане на решение, както е показано в Таблица 2.6.

Таблица 2.6. Матрица за вземане на решение при модификация, формализираща коефициенти за предимство на алтернативите.

Критерии		**Тегла за важност на критериите**			**Алтернативи – оценки и коефициенти**						
					$\boldsymbol{A_1}$			…	$\boldsymbol{A_I}$		
Тип	**Критерий**	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$	…	ЛВР$^{\mathbf{1}}$	…	ЛВР$^{\boldsymbol{K}}$
За оценяване на алтернативите	C_1	w_1^1	…	w_1^K	$a_{1,1}^1$	…	$a_{1,1}^K$	…	$a_{I,1}^1$	…	$a_{I,1}^K$
	…	…	w_j^k	…	…	$a_{i,j}^k$	…	…	…	$a_{i,j}^k$	…
	C_J	w_J^1		w_J^K	$a_{1,J}^1$	…	$a_{1,J}^K$	…	$a_{I,J}^1$	…	$a_{I,J}^K$
За предимство на алтернативите	C_1	-			ψ_1^1	ψ_1^k	ψ_1^K	…	ψ_I^1	ψ_I^k	ψ_I^K
	…							ψ_i^k			
	C_N							…			

2.4. Изводи

Предложените модификации на метода на претеглената сума и метода на претегленото произведение въвеждат три новости: 1) отчитане на персоналната компетентността на ЛВР; 2) генериране на оценки за алтернативите и 3) прилагане на допълнително предимство на алтернативите спрямо ключови критерии. Всички те, могат да бъдат съчетани в един обобщен алгоритъм и математически модел, включващи едновременното прилагане и на трите модификации или на различни комбинации между тях.

Модификациите могат да се разглеждат като съществена стъпка при реализирането на система за подпомагане вземането на решения. Функционираща в уеб среда, такава система би допринесла значително за автоматизиране на някои процеси и улесняване на ЛВР в условията на групово вземане на решения. За реализирането на такъв инструмент, е необходимо освен наличието на модели и алгоритми, така и проектирана подходяща софтуерна архитектура.

ГЛАВА 3.
ПРОЕКТИРАНЕ НА СОФТУЕРНИ АРХИТЕКТУРИ НА СИСТЕМИ ЗА ПОДПОМАГАНЕ ВЗЕМАНЕТО НА РЕШЕНИЯ

В настоящата глава са описани предложените и проектирани две софтуерни архитектури, отчитайки специфичните особености на СПВР. Проектирането на две архитектури за една и съща система е обосновано от необходимостта за определяне на подходящ набор от технологии за разработка и експлоатационна среда, като се вземат предвид технологичните изисквания, очакваната натовареност и контекста на използване.

Като отправна точка за проектирането са използвани описаните в Глава 2 модификации, на база на които допълнително са формулирани комбиниран и обобщен математически модел. Тези модели определят ядрото на бизнес логиката за вземане на решение. За целта е създаден обобщен алгоритъм за прилагане на математическите модели, като ключови стъпки от него са описани с псевдокод. Функционалността, която СПВР трябва да притежава е проектирана чрез адаптирани UML диаграми. Дефинирани са роли, функционални изисквания и случаи на употреба на системата. Описани са необходимите формуляри и последователността на взаимодействие между потребителя и системата. В края на настоящата глава е направено сравнение на силните страни и ограниченията на двата архитектурни подхода спрямо ключови характеристики.

3.1. Математически модели на комбинираните модификации на метода на претеглената сума

Комбинирайки предложената модификация (2.1) – (2.4), отчитаща компетентност на ЛВР за всеки критерий поотделно чрез теглови коефициенти заедно с модификацията (2.13) – (2.21), формализираща генериране на оценки за алтернативите, се предлага следния комбиниран модел на модифицирания метод на претеглената сума:

$$A_{i\ CombinedWSM}^{GDM} = \sum_{k=1}^{K}\sum_{j=1}^{J} \lambda_j^k w_j^k \beta_{i,j}^k \tag{3.1}$$

При ограничения (2.2), (2.3) и следното допълнително условие:

$$\beta_{i,j}^k = \begin{cases} a_{i,j}^k & a \in [0,1] \\ (b_{i,j}^k)^{norm} & b \in [0,1] \\ c_{i,j}^k & c \in [0,1] \end{cases} \tag{3.2}$$

За генерирането на $(b_{i,j}^k)^{norm}$ и $c_{i,j}^k$, са валидни формулировките (2.17) и (2.18), както и ограниченията (2.19) – (2.21).

При тази формулировка се използва въведената система от означения в Глава 2, с допълнението, че $A_{i\ CombinedWSM}^{GDM}$ изразява агрегираната оценка на i-*mata* алтернатива, изчислена по комбинирания модел на модифицирания метод на претеглената сума.

Съчетавайки комбинацията от (3.1) с модификацията, формализираща коефициент за предимство на алтернативите, се предлага следния обобщен модел на модифицирания метод на претеглената сума:

$$A_{i\ GeneralizedWSM}^{GDM} = \sum_{k=1}^{K}\left(\psi_i^k \sum_{j=1}^{J} \lambda_j^k w_j^k \beta_{i,j}^k\right) \tag{3.3}$$

където:

— $A_{i\ GeneralizedWSM}^{GDM}$ изразява агрегирана оценка на i-*mata* алтернатива, изчислена по обобщения модел на модифицирания метод на претеглената сума.

За генерирането на ψ_i^k, е валидно условието (2.23), както и ограниченията (2.24) – (2.29).

Отчитайки възможността за групиране на критериите с цел подпомагане на администрирането, задавайки общ коефициент за компетентност на ЛВР спрямо група критерии, комбинирания модел и обобщения модел придобиват следните форми:

$$A_{i\ CombinedWSM}^{GDM} = \sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{m_l=1}^{M_l} \lambda_l^k w_{m_l}^k \beta_{i,m_l}^k \tag{3.4}$$

$$A_{i\ GeneralizedWS}^{GDM} = \sum_{k=1}^{K}\left(\psi_i^k \sum_{l=1}^{L}\sum_{m_l=1}^{M_l} \lambda_l^k w_{m_l}^k \beta_{i,m_l}^k\right) \tag{3.5}$$

При ограничения (2.8), (2.9) и следното допълнително условие:

$$\beta_{i,m_l}^k = \begin{cases} a_{i,m_l}^k & a \in [0,1] \\ (b_{i,m_l}^k)^{norm} & b \in [0,1] \\ c_{i,m_l}^k & c \in [0,1] \end{cases} \tag{3.6}$$

За така предложените комбиниран модел и обобщен модел на модифицирания метод на претеглената сума, са валидни всички обозначения и ограничения, дефинирани за отделните модифицирани модели от Глава 2.

Матрицата за вземане на решение, върху която се прилага обобщения модел, съчетаващ всички предложени модификации над WSM (3.4), има вида, показан в Таблица 3.1:

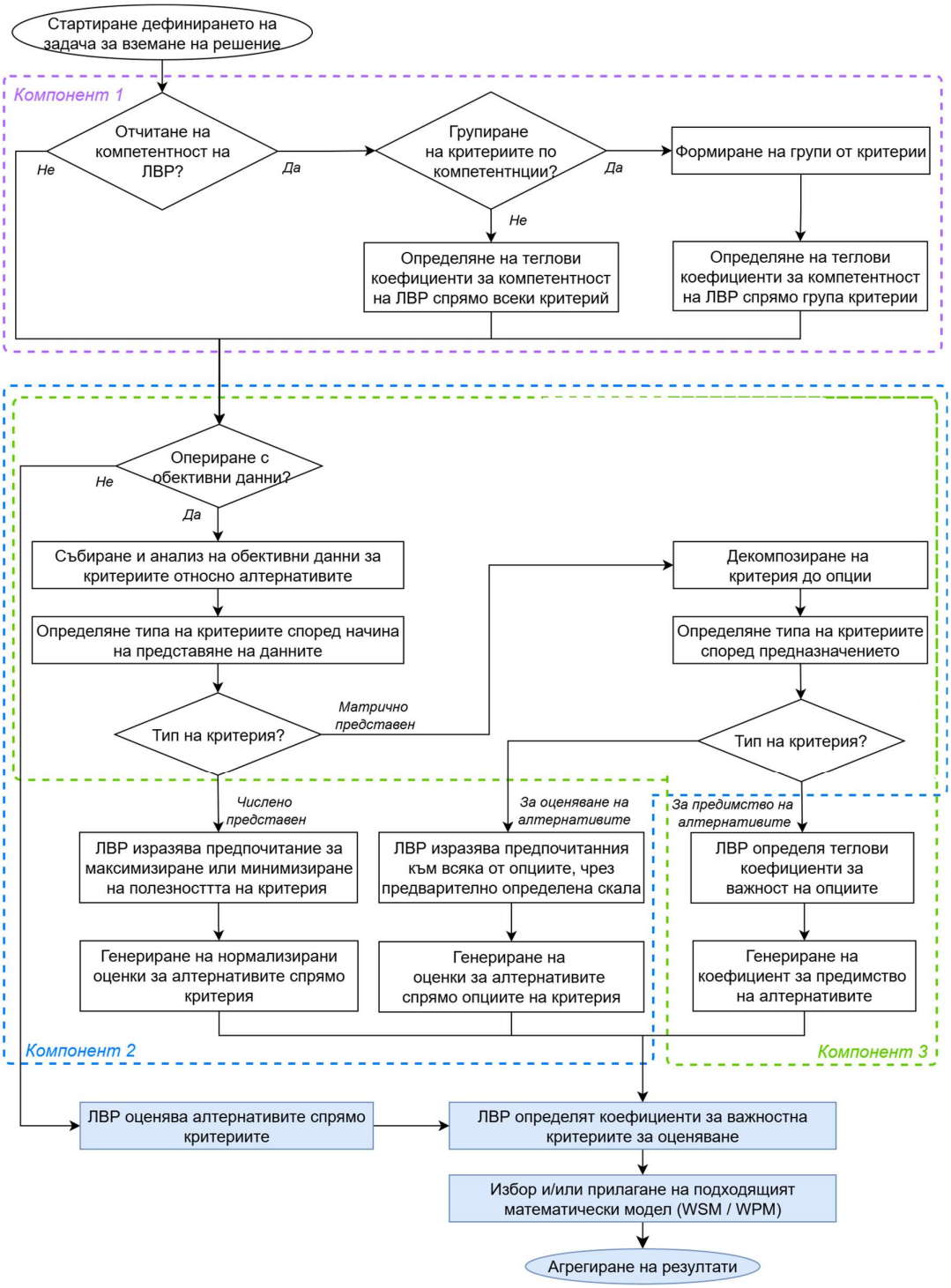
Таблица 3.1. Матрица за вземане на решение при обобщен модел на модифицирания метод на претеглената сума.

Критерии			Тегла за компетентност на ЛВР			Полезност на критериите при нормализация			Тегла за важност на критериите			Алтернативи – субективни оценки, генерирани оценки, коефициенти						
Тип 1	Тип 2	Критерий										A_1			...	A_I		
			ЛВР^1	...	ЛВР^K	ЛВР^1	...	ЛВР^K	ЛВР^1	...	ЛВР^K	ЛВР^1	...	ЛВР^K	...	ЛВР^1	...	ЛВР^K
За оценяване	субективно оценявани	C_1	λ_1^1	λ_j^k	λ_1^K	-			w_1^1	...	w_1^K	$a_{1,1}^1$	...	$a_{1,1}^K$	...	$a_{1,1}^1$	...	$a_{1,1}^K$
		...	...		...				...	...	...	...	$a_{i,j}^k$	...	...	...	$a_{i,j}^k$	...
		C_j	...		...				...	...	...	$a_{1,j}^1$	...	$a_{1,j}^K$	...	$a_{I,j}^1$	...	$a_{I,j}^K$
	числено представени	C_j	...		...	max/min	...	max/min	...	...	...	$b_{1,j}^1$	...	$b_{1,j}^K$	...	$b_{I,j}^1$	...	$b_{I,j}^K$
		...	...		...	...	...	...	...	w_j^k	...	...	$b_{i,j}^k$	...	...	...	$b_{i,j}^k$	...
		C_j	...		...	max/min	...	max/min	...	...	...	$b_{1,j}^1$	...	$b_{1,j}^K$	...	$b_{I,j}^1$	...	$b_{I,j}^K$
	матрично представени	C_j	...		...	-			...	...	...	$c_{1,j}^1$	...	$c_{1,j}^K$	...	$c_{I,j}^1$	...	$c_{I,j}^K$
		...	...		...				...	...	...	...	$c_{i,j}^k$	...	...	...	$c_{i,j}^k$	...
		C_J	λ_J^1		λ_J^K				w_J^1	...	w_J^K	$c_{1,J}^1$	...	$c_{1,J}^K$	...	$c_{I,J}^1$	...	$c_{I,J}^K$
За предимство		C_n	-			-						ψ_1^1	...	ψ_1^K	ψ_i^k	ψ_I^1	...	ψ_I^K
		...																
		C_N																

Върху така дефинираните математически модели на комбинирания и обобщения метод на претеглената сума, се съставя обобщен алгоритъм, на който се базира бизнес логиката на прототипа на СПВР.

3.2. Обобщен алгоритъм, реализиращ комбиниране на модификациите

Обобщеният алгоритъм има за цел да интегрира всички предложени в настоящия дисертационен труд модификации върху методите WSM и WPM в една система за подпомагане вземането на решения. Алгоритъмът е представен на Фиг. 3.1.



Фиг. 3.1. Обобщен алгоритъм, реализиращ комбиниране на предложените модификации.

В контекста на софтуерното инженерство, от обобщения алгоритъм се обособяват три основни компонента, съответстващи на всяка от предложените модификации. Логиката за комбиниране на модифицираните математически модели, според архитектурния подход и използваните технологии, може да се реализира на различни програмни езици. Затова като отправна точка е представен псевдокод, който дефинира входните параметри, основните структури от данни и примерна валидация. Той е валиден за всички последващи блокове от псевдокод и е илюстриран на Фиг. 3.2.

```
// Входни параметри:
 const I        // брой на алтернативите
 const K        // брой на ЛВР
 const J        // брой на критериите за оценяване
 const L        // брой на групите от критерии
 const N        // брой на критериите за предимство
 const Rj, Rn   // брой на опциите в j-тия и n-тия критерий
// Основни работни структури от данни:
 var A[I]                // агрегирани стойности за всички алтернативи
 var w[K][J]             // важност на критерий j от ЛВР k
 var λ[K][J], λ[K][L]    // компетентност на ЛВР k по критерий j и група l
 var β[I][J][K]          // оценка на алтернатива i по критерий j от ЛВР k
 var Ψ[K][I]             // генериран коефициент за предимство
Begin   // Валидация на данните:
// Проверка дали коефициентите за важност се сумират до 1
  for k = 1 to K do
    var sum_w = 0
    for j = 1 to J do
      sum_w = sum_w + w[k][j]
    end for
    if sum_w == 1 then
      print "Грешка: теглата на ЛВР {k} не сумират до 1."
      return
  end for
...
End
```

Фиг. 3.2. Блок псевдокод, дефиниращ входните параметри, работните структури от данни и валидацията на входните данни.

Предложения псевдокод следва нотацията описана в математическите модели, включително имена на променливи, броячи и други означения. На същия принцип като в примерната валидация от Фиг. 3.2, се валидират всички останали ограничения наложени от математическите модели.

Компонент 1 капсулира модификацията, отчитаща компетентността на ЛВР. Разграничават се три варианта на определяне на коефициентите за компетентност:

- за всеки критерий поотделно;
- за групи от критерии с обща стойност на коефициента за компетентност;
- глобално за задачата, което представлява частен случай на групирането на критерии, когато групата е само една ($L = 1$).

Псевдокод, реализиращ логиката на Компонент 1 е представен на Фиг.3.3.

```
Begin Component1 //Компонент1 – модификация, отчитаща компетентността на ЛВР
  if λ == true then
    if λ[k][j] == true then   // Компетентност по критерии j
       for k = 1 to K do
          for j = 1 to J do
             print "Попълнете компетентността по критерий {j} на ЛВР {k}"
             λ[k][j] = input
          end for
          Проверка дали коефициентите за компетентност се сумират до 1
       end for
       return λ[k][j]
    else if λ[k][l] == true then  // Компетентност по групи l
       if L == 1 then    // Компетентност глобално за задачата при l=1
          for k = 1 to K do
            print "Попълнете компетентността на ЛВР {k}"
            λ[k] = input
            end for
          Проверка дали коефициентите за компетентност се сумират до 1
          return λ[k]
       for k = 1 to K do
          for l = 1 to L do
            print "Попълнете компетентността по група {l} на ЛВР {k}"
            λ[k][l] = input
          end for
          Проверка дали коефициентите за компетентност се сумират до 1
       end for
       return λ[k][l]
  return λ = null
End
```

Фиг. 3.3. Блок псевдокод, описващ логиката на Компонент 1 в СПВР.

Компонент 2 и **Компонент 3** преизползват общи модули, например за декомпозиране на критериите до опции и типизиране на критериите. В тази връзка критериите се типизират по два признака – според предназначението и според начина, по който се формира оценка. Според предназначението си критериите могат да бъдат:

- за оценяване на алтернативи;
- за предимство на алтернативи.

Според начина за формиране на оценка, се разграничават следните три типа:

- субективни;
- числено представени;
- матрично представени (декомпозируеми до опции).

При субективните критерии оценките се формират пряко от предпочитанията на ЛВР. При числено и матрично представените критерии оценките се генерират въз основа на обективни данни и изразените предпочитания на ЛВР. Модул за типизиране на критериите, е представена чрез псевдокод на Фиг. 3.4.

```
Begin Criteria Classification    // Логиката за типизиране на критериите
  for j = 1 to J do
    print "Определете типа на критерий {j}"
    β[j] = input  // Субективен / Числен / Матричен
    if β[j] == a then    // "a" е тип субективни критерии
       Подготовка на формуляр за събиране на субективни оценки от ЛВР
    else if β[j] == b then   // "b" е тип числено представени критерии
       for i = 1 to I do
           β[j] = input // Попълване на стойностите на критериите
           Подготовка на формуляр за събиране на cost/benefit предпочитания
       end for
    else if β[j] == c then   // "c" е тип матрично представени критерии
       Декомпозиране до опции
       print "Определете типа на матрично представения критерий {j}"
       β[j] = input  // За генериране на оценка / За генериране на коефициент
       if β[j] == Ψ then
           J--      // Изваждане от масива от критерии за оценяване
           N++      // Добавяне към масива от критерии за предимство
       end if
       if J > 1 then
           Подготовка на формуляр за събиране на предпочитания към опции Rj
       if N > 1 then
           Подготвяне на формуляр за събиране на предпочитания към опции Rn
    end if
  end for
End
```

Фиг. 3.4. Блок псевдокод, описващ модула за типизиране на критериите.

Подготовката на индивидуален формуляр за събиране на оценки и предпочитания от ЛВР, е процес, който се реализира от компонента за генериране на формуляри. Това е спомагателен компонент, който капсулира функционалността по създаване на потребителски интерфейс по зададени параметри, като брой различни атрибути, например текстови полета за описание, радио-бутони и т.н.

Операцията „Декомпозиране до опции“ е обособена в отделен модул, като псевдокод за него, е представен на Фиг. 3.5.

```
Begin Criteria Decomposition  // Модул за декомпозиране на критериите до опции
   for j = 1 to J do
       for i = 1 to I do
           for r = 1 to Rj do
               v[i][j][r] = input // Попълване на стойностите на опциите
           end for
       end for
   end for
   if J >= 1 then
       Определяне на скала за оценяване важността на опциите
   for n = 1 to N do
       for i = 1 to I do
           for r = 1 to Rn do
               v[i][n][r] = input // Попълване на стойностите на опциите
           end for
       end for
   end for
End
```

Фиг. 3.5. Блок псевдокод, описващ модула за декомпозиране на критериите до опции.

Въз основа на нововъведения подход за обработка на критериите (типизиране и декомпозиране) се стартират функции по:

- генериране на оценки за алтернативите по числено представени критерии;
- генериране на оценки за алтернативите по матрично представени критерии;
- генериране на коефициенти, даващи предимство на алтернативите.

Генерирането на оценки спрямо числено представените критерии (2.17), е обособено в самостоятелен модул описан чрез псевдокода, представен на Фиг. 3.6.

```
Begin Generation of Alternative Scores type  b  // Модул за генерирането на
оценки спрямо числено представените критерии
  function normalization_technique
  var b_value[I][J]
  var b_cost_benefit_preference[J][K]
  var b_generated[I][J][K]
  for i = 1 to I do
        for k = 1 to K do
            for j = 1 to J do
                b_generated[i][k][j] = normalization_technique(
                                       b_value[i][j],
                                       b_cost_benefit_preference[j][k])
            end for
        end for
  end for
  return β[I][J][K] = b_generated[I][J][K]
End
```

Фиг. 3.6. Блок псевдокод, описващ генерирането на оценки спрямо числено представените критерии.

Генерирането на оценки (2.18) и генерирането на коефициенти (2.23) спрямо матрично представените критерии използват идентични структури от данни. Двата процеса също така преизползват функционалността на модула за декомпозиране, представен на Фиг. 3.5. Поради това може да се вземе архитектурно решение те да бъдат обединени в единен модул. Въпреки различния характер на генерираните резултати, това обединение води до подобряване на програмния код и улеснява интерпретирането и имплементацията на обобщения алгоритъм. Псевдокодът на модула е представен на Фиг. 3.7.

```
 Begin Generation of Scores type c and Ψ Coefficients // Модул за генерирането
на оценки и коефициенти спрямо матрично представените критерии
  var scale_max
  var c_option_values[I][J][Rj] //Стойности на опциите - критериите за оценка
  var c_option_preference[K][J][Rj] //Предпочитания към опциите
  var Ψ_option_values[I][N][Rn] //Стойности на опциите – критерии за предимство
  var Ψ_option_preference[K][N][Rn] //Предпочитания към опциите
  var c_generated[I][J][K] //Генерирани оценки
  var Ψ_generated[I][K] //Генерирани коефициенти
  var sum, prod //Работни променливи
  for i = 1 to I do
    for k = 1 to K do
      if β[j] == c then
        for j = 1 to J do
          for rj = 1 to Rj do
           sum  =  sum+c_option_preference[k][j][rj]*c_option_values[i][j][rj]
          end for
          c_generated[i][j][k] = sum / scale_max * Rj
        end for
      if β[j] == Ψ then
        for n = 1 to N do
          for rn = 1 to Rn do
            sum = sum+Ψ_option_preference[k][n][rn]*Ψ_option_values[i][n][rn]
          end for
          prod = prod * sum
        end for
        Ψ_generated[i][k] = prod
    end for
  end for
  if β[j] == c then
  return β[I][J][K] = c_generated[I][J][K]
  if β[j] == Ψ then
  return Ψ_generated[I][N][K]
 End
```

Фиг. 3.7. Блок псевдокод, описващ генерирането на оценки и коефициенти спрямо матрично представените критерии.

Потокът на изпълнение в рамките на обобщения алгоритъм определя кой от предложените в Глава 2 и Глава 3, т. 3.1 модифицирани математически модели може да бъде приложен за изчисление на агрегираните резултати. Логиката за филтриране и предоставяне на избор на подходящ за задачата модел, е описана чрез псевдокода, представен на Фиг. 3.8.

```
Begin Mathematical Model Reduction & Selection    // Избор от допустимите модели
 if λ == true && β == true && Ψ == true then
   if L == true then
        Допустим модел е WSM (3.5)
   else
        Допустим модел е WSM (3.3)
 if λ == true && β == true && Ψ == false then
   if L == true then
        Допустим модел е WSM (3.4)
   else
        Допустим модел е WSM (3.1)
 if λ == true && β == false && Ψ == true then
   if L == true then
        За бъдеща разработка
   else
        За бъдеща разработка
 if λ == true && β == false && Ψ == false then
   if L == true then
        Избор между WSM модел (2.7) и WPM модел (2.11)
   else
        Избор между WSM модел (2.1) и WPM модел (2.5)
 if λ == false && β == true && Ψ == true then
   За бъдеща разработка
 if λ == false && β == true && Ψ == false then
   Допустим модел е WSM (2.13)
 if λ == false && β == false && Ψ == true then
   Избор между WSM модел (2.22) и WPM модел (2.30)
 if λ == false && β == false && Ψ == false then
   Избор между класическите: WSM модел (1.3) и WPM модел (1.7)
End
```

Фиг. 3.8. Блок псевдокод, описващ компонента за филтриране и избор на подходящ за задачата модел.

Различните комбинации между компонентите позволяват резултатите да се агрегират по един от 13 различни модела – 9 с метода WSM и 4 с WPM, както показват референциите от Фиг. 3.8. В случаите, когато дефинирането на задачата не предполага прилагане на предложените модификации, алгоритъмът преминава към избор на математически модел между класическите методи (1.3) и (1.7).

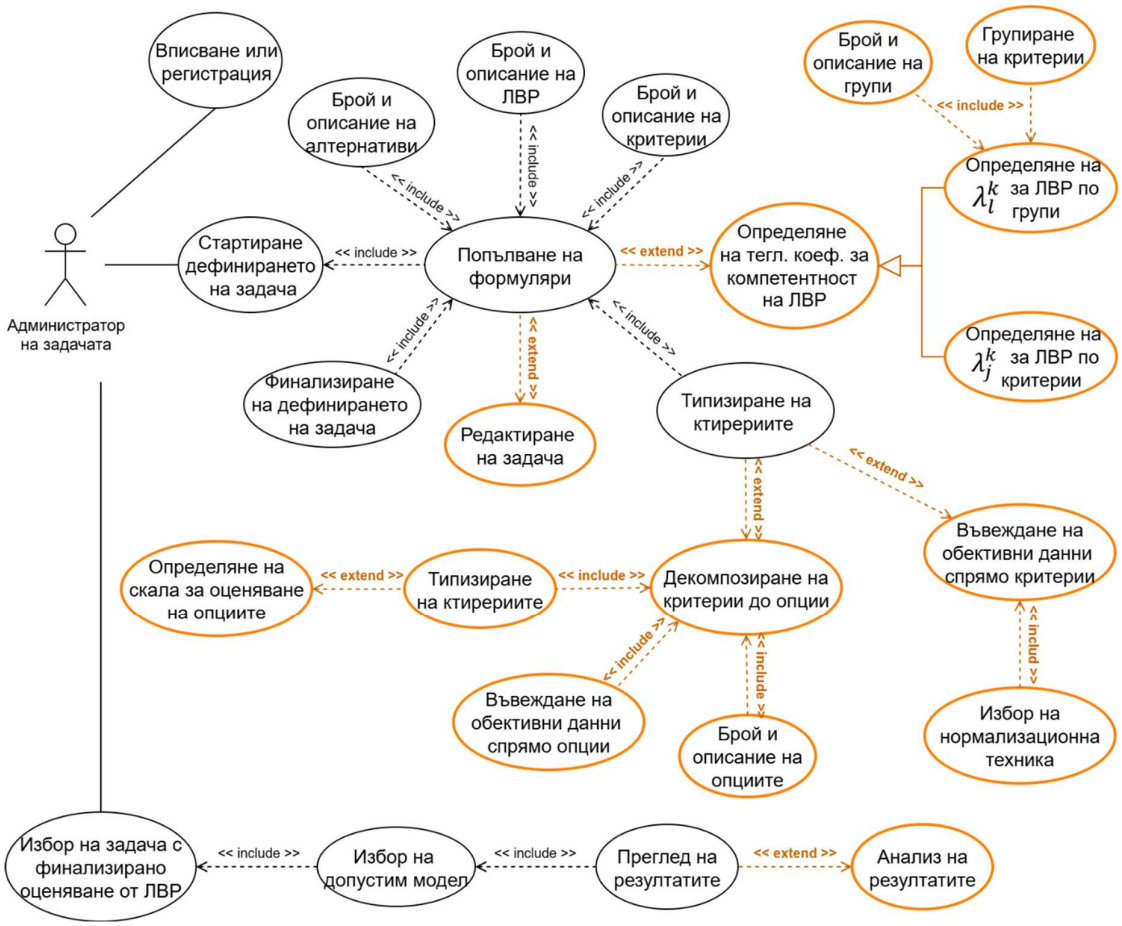
Предложеният алгоритъм е ядрото на бизнес логиката на СПВР. Въпреки това, за проектирането на пълноценно приложение работещо в уеб среда, е необходимо да се дефинират и изискванията за функционалност и взаимодействие с потребителя.

3.3. Проектиране на функционалността на СПВР

В следствие от предварително направен анализ на изискванията, се дефинират всички функции, които СПВР покрива, както и правилата за изпълнението им. Идентифицират се всички заинтересовани страни, техните роли и необходимия потребителски интерфейс. Разграничават се три роли:

- Администратор на задача за вземане на решение;
- Лица, вземащи решение;
- Системна роля, която обединява дейностите, изпълнявани автоматизирано от системата, следвайки проектираната логика и предложените алгоритми.

Ролята на администратора и взаимодействието му със СПВР, е илюстрирано чрез диаграма на случаите на употреба (UML Use Case Diagram), представена на Фиг. 3.9.



Фиг. 3.9. Адаптирана UML Use Case диаграма за ролята на администратор на задача.

След като се впише в системата, администраторът може да я използва по два основни сценария: (1) стартиране на процес по дефиниране на задача, което включва попълване на формуляри за пълното ѝ описание или (2) достъпване на вече съществуваща задача, оценена от всички участници (ЛВР), с цел прилагане на избран математически модел и агрегиране на резултати. Допълнителните функционалности, които разширяват тези базови сценарии, отразяват прилагането на предложените модификации към класическите методи WSM и WPM.

Ролята на ЛВР е описана с аналогична диаграма, представена на Фиг. 3.10.

Фиг. 3.10. Адаптирана UML Use Case диаграма за ролята на ЛВР.

След вписване в системата, дейностите на ЛВР се свеждат до попълване на формуляри, свързани с предварително дефинираните от администратора елементи на задачата. Както е посочено на Фиг. 3.10, някои от формулярите са задължителни, докато други се генерират условно в зависимост от избраните сценарии за употреба, които отразяват предложените в дисертационния труд модификации.

В индивидуални задачи участва едно лице, а в групови – множество ЛВР. Всеки от тях трябва да въведе оценки и предпочитания независимо от физическото си местоположение, от времето на участие, както и независимо от останалите участници. Тези изисквания за децентрализирана работа на участниците, налага използването на уеб базирана среда, която осигурява гъвкав достъп.

Автоматизираните операции, изпълнявани от СПВР, са представени на Фиг. 3.11.

Фиг. 3.11. Адаптирана UML Use Case диаграма за системната роля.

Важно е да се отбележи, че системната роля изпълнява самостоятелни операции, които не следват последователна логика, а имат подпомагаща функция спрямо действията на останалите роли в СПВР.

Автентикацията като механизъм за удостоверяване на самоличността на потребителите, може да бъде реализирана следвайки добрите практики [Barkadehi и др., 2018] или да бъде изнесена като външна услуга, например Google Identity Services.

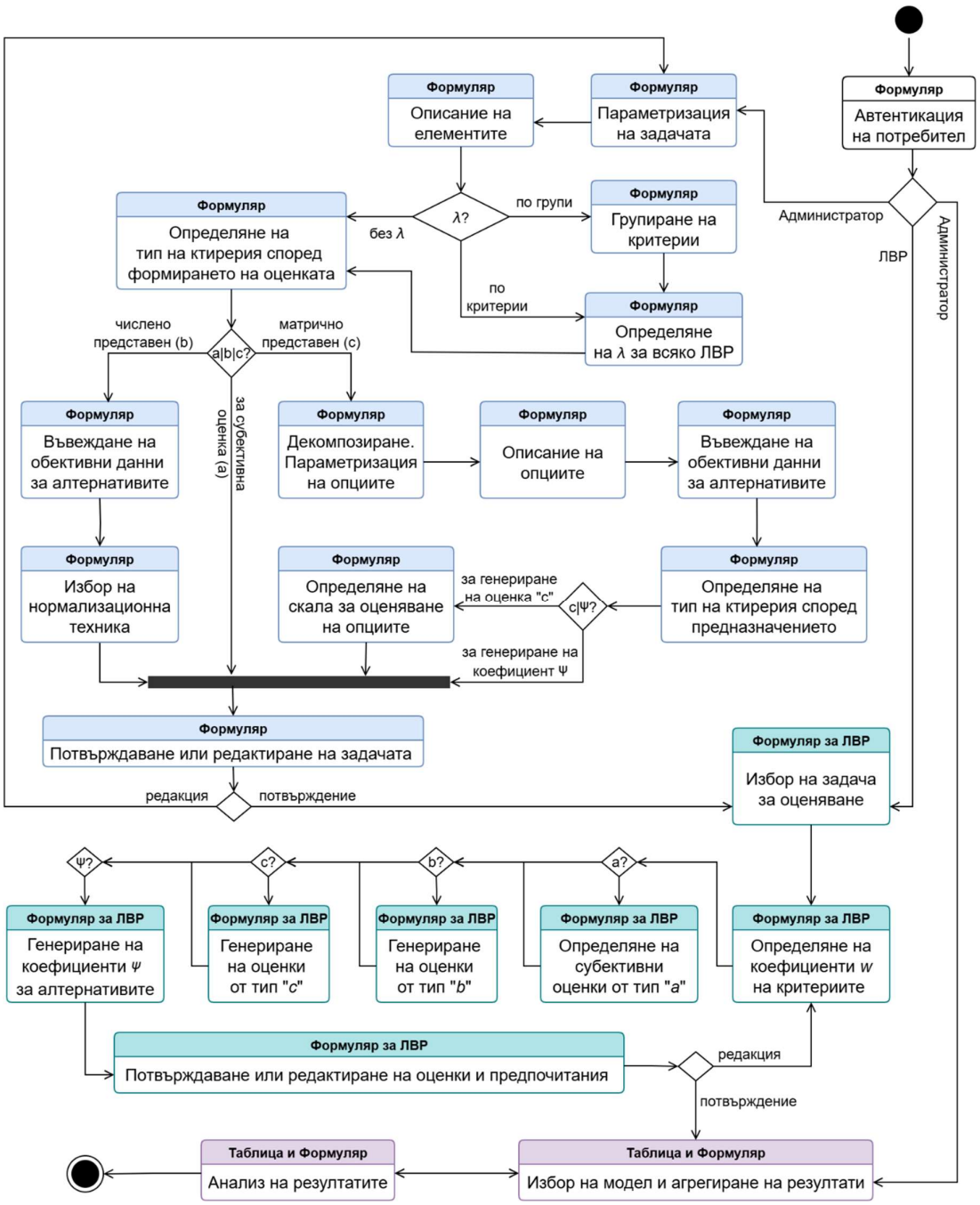
Валидация на входните данни се изпълнява в две стъпки. При слоя на потребителския интерфейс се проверява коректността на типа и форма́та на данните с регулярни изрази (Regular Expressions). В слоя на бизнес логиката се прилагат ограниченията, произтичащи от математическите модели, като примерен подход е предложен чрез псевдокода на Фиг. 3.2.

Генерирането на коефициенти и оценки спрямо матрично и числено представените критерии е специфична функционалност за СПВР и е представена с псевдокода на Фиг. 3.6 и Фиг. 3.7. Същото важи и за определянето на допустим модел за агрегиране на резултатите, което е представено с псевдокода от Фиг. 3.8.

Изчисляването на агрегираните резултати по един от допустимите 13 варианта, включва и междинни изчисления, с цел да се подпомогне анализа на резултатите. Преобразуването на резултатите допълнително подпомага анализа, като привежда резултатите в процентна форма, в случая на WPM е необходимо да бъде направена предварителна линеаризация.

Записът и извличането на данни от базата данни са ключови операции, които съпътстват действията на ролите в СПВР и служат като връзка между слоя за бизнес логика и постоянното хранилище на системата.

Динамично генерираните формуляри в СПВР, структурират потребителското взаимодействие, а преходите между тях дефинират логиката на работа със системата. Това е показано с адаптирана диаграма на състоянията (UML State Machine) на Фиг.3.12.



Фиг. 3.12. Адаптирана UML State Machine диаграма на потребителския интерфейс.

Динамично генерираните формуляри се създават в реално време, докато потребителят работи със системата. Това означава, че формулярът не е предварително подготвен (статичен), той се генерира според зададени условия, параметри и контекст на конкретната задача. Процесът е **интерактивен**, защото системата реагира на действията и избора на потребителя отново в реално време, като адаптира съдържанието и структурата на формуляра. Процесът е и **паралелен**, тъй като множество потребители могат едновременно да създават и попълват формуляри с различни параметри от собствените си акаунти, без това да води до конфликти или загуба на данни.

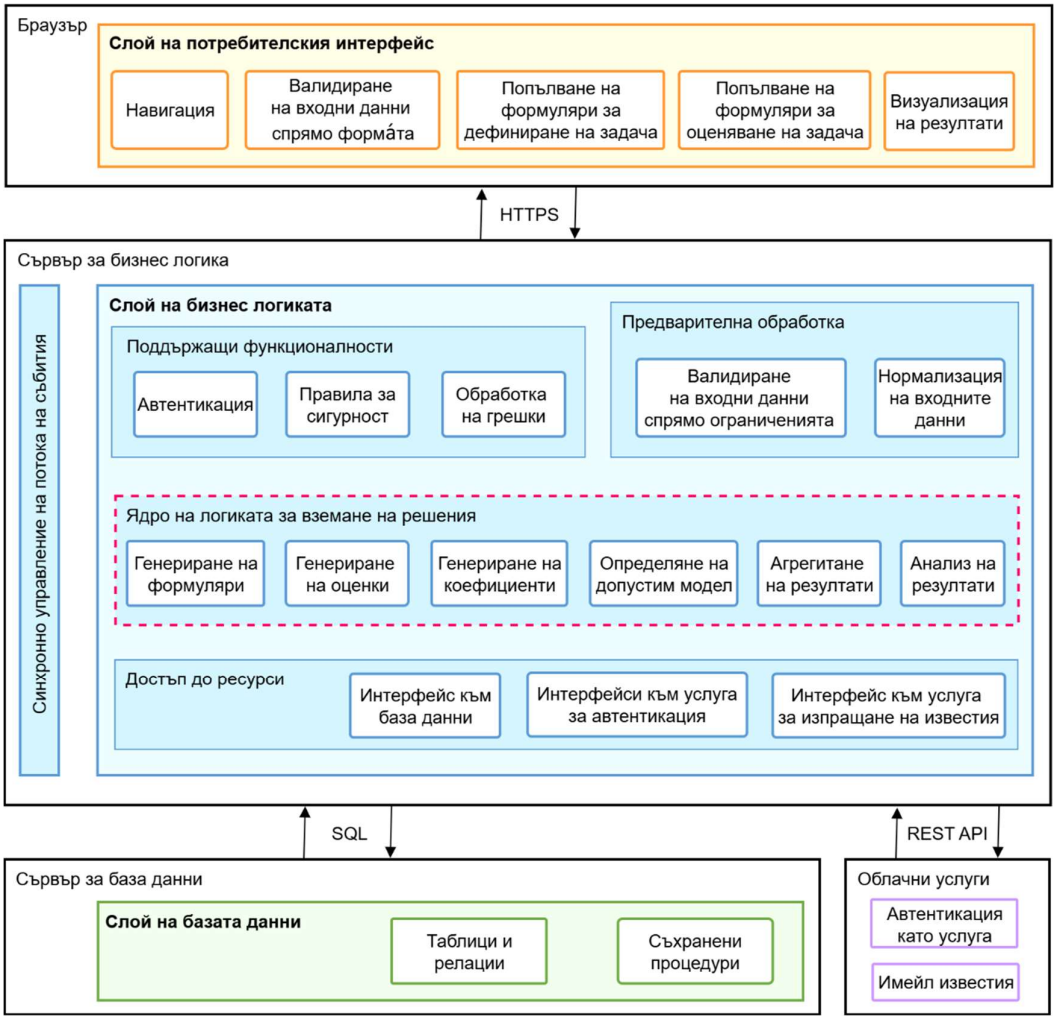
От гледна точка на дефинирането на задача за вземане на решение, формулярите и условните преходи между тях имат за цел единствено да подпомогнат администратора при структурираното и описанието ѝ. Както е посочено в Глава 2, бизнес анализът не е обект на настоящото изследване. Това означава, че системата не е предназначена да подпомага етапите на анализ на проблема или събиране на изходна информация. Администраторът трябва да разполага с подготвени и проверени данни преди да започне процеса по дефиниране на задачата.

Преходите между формулярите отразяват основната бизнес логика и последователността на взаимодействие между системата и потребителя, което е другият ключов аспект, илюстриран на Фиг. 3.12. Основният принцип, е че винаги преходът към следващия формуляр се прави след като текущият бъде попълнен и потвърден от потребителя. При всеки такъв преход системата изпълнява две основни операции: (1) съхраняване на въведените данни в базата данни и/или (2) извличане на необходимата информация за създаване на следващия формуляр. По този начин се осигурява надеждност и проследимост на въведените данни, както и условие за редактиране на вече попълнен формуляр.

3.4. Уеб базирана СПВР с монолитна трислойна архитектура

Предложената трислойна архитектура (three-tier architecture), следва класическия трислоен архитектурен шаблон, който осигурява ясно разделение на отговорностите и инфраструктурата, улеснява поддръжката и налага структуриран подход към развитието на системата. От гледна точка на структурирането и разгръщането на кода,

е избрано да се следва монолитен архитектурен стил, а потокът на данните се управлява синхронно. Топология на трислойната архитектура използвана за разработка на прототип на СПВР, е дадена на Фиг. 3.13 [Dimitrova и др., 2021].



Фиг. 3.13. Топология на проектираната трислойна архитектура на СПВР.

Слоят на потребителският интерфейс (UI), наричан още презентационен слой се изпълнява от уеб браузъра на потребителския компютър. Чрез него се осъществява цялото взаимодействие между хората и СПВР, което включва навигация между различните формуляри, попълването и валидирането на форма́та на въведените данни. Технологиите използвани на това архитектурно ниво са HTML и CSS, като са спазени принципите за адаптивен уеб дизайн.

Уеб страниците на потребителския интерфейс се генерират динамично на сървъра за бизнес логика и се предават към клиентския браузър посредством

протокола HTTPS. От гледна точка на обхват на функционалностите, **слоят на бизнес логиката** може да се раздели на четири функционални области:

- Ядрото на логиката обхваща ключовите алгоритми и функционалности, свързани с вземането на решения, като реализацията на класическите и модифицирани модели на методите WSM и WPM, както и определянето на допустим модел, агрегирането и анализа на резултатите;
- Поддържащите функционалности, не са част от основните алгоритми, но осигуряват тяхната работа (автентикация, обработка на грешки, правила за сигурност и др.);
- Предварителната обработка, се изразява в нормализация и валидация на входните данни спрямо ограниченията на моделите за вземане на решения, преди те да бъдат използвани от ядрото на логиката;
- Достъпа до ресурси управлява комуникацията с вътрешни ресурси като базата данни, чрез подготовка и изпращане на подходящи SQL заявки. Освен това използва API интерфейси за комуникация с външни услуги като автентикация, например с Google Identity Services.

Прототипът на СПВР, базиран на трислойна архитектура, е разработен използвайки софтуерния пакет XAMPP, който включва уеб сървър Apache с интерпретатор за PHP, както и сървър за база данни със система за управление на база данни (СУБД).

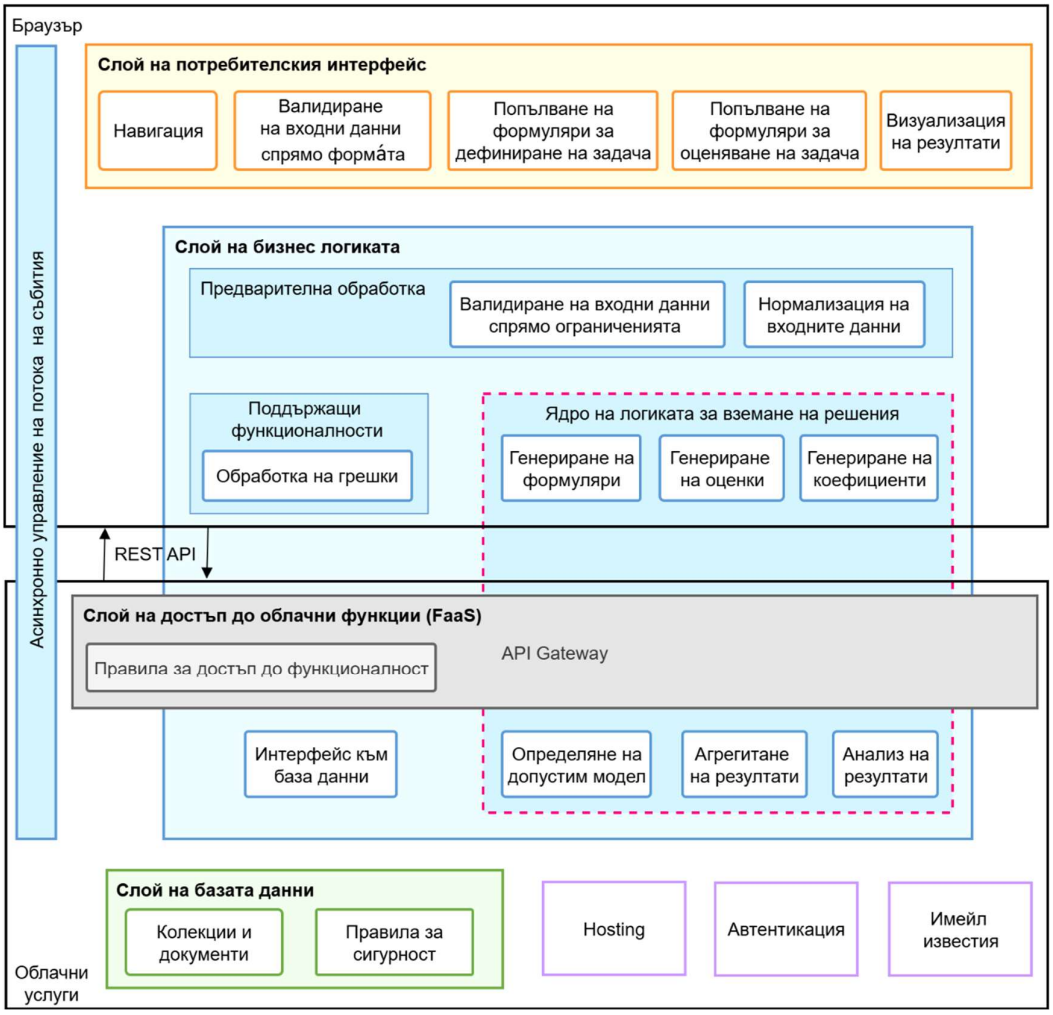
Слоят за база данни абстрахира достъпа и улеснява управлението на данни за формуляри, оценки, критерии, алтернативи и ЛВР. В прототипа се използва релационната база данни MySQL, част от пакета XAMPP. Комуникацията със сървъра за база данни се осъществява чрез собствения протокол на СУБД, върху TCP/IP мрежова връзка, с поддръжка на SSL/TLS криптиране за защита на данните. Този подход осигурява бърз и надежден обмен на SQL заявки и резултати между слоя за бизнес логика и слоя за база данни.

Важно е да се отбележи, че трислойната архитектура използва синхронна обработка на събитията. Това означава, че взаимодействието между системата и потребителя, както и между отделните компоненти на системата се извършва в строго

определена последователност, при която всяка стъпка изчаква завършването на предишната, преди да продължи.

3.5. Уеб базирана СПВР с безсървърна архитектура

Предложената безсървърна архитектура използва архитектурния шаблон на разпределени функции (FaaS), които се активират при конкретни действия на потребителя или вътрешни за системата събития – асинхронно управление на потока на събитията. Този подход осигурява висока мащабируемост, минимална необходимост от управление на хардуерна и софтуерна инфраструктура, както и много висока адаптивност спрямо натоварването на системата. Топология на безсървърната архитектура, на която се базира разработен прототип на СПВР е показана на Фиг.3.14 [Dimitrova и др., 2024, a].



Фиг. 3.14. Топология на проектираната безсървърна архитектура на СПВР.

При безсървърната архитектура отговорностите също са структурирани и разделени на слоеве, но няма постоянни, специализирани сървъри за всеки слой, а разделението е изцяло логическо и не зависи от физическа инфраструктура.

Слоят на потребителския интерфейс е реализиран чрез React приложение, което се изпълнява в уеб браузъра. Чрез компонентен подход, формулярите и визуалните елементи са капсулирани, а навигацията между отделните изгледи се управлява от React Router. Потребителското взаимодействие, изразяващо се в попълването на различните формуляри, визуализацията на резултатите както и валидирането на входните данни също се изпълнява на клиентската машина. Технологиите използвани в слоя за потребителски интерфейс са CSS и React, а чрез JSX описание на елементите на уеб страниците се генерира HTML. Комуникацията между React приложението и облачните функции се осъществява чрез REST API използващо HTTPS.

Слоят за бизнес логика е разпределен между React приложението и облачните функции на Firebase (Firebase Cloud Functions). Цялата предварителна обработка, изразяваща се в нормализация на входните данни и валидацията им спрямо ограниченията се извършва в браузъра. Там се изпълнява и обработката на грешки. Докато за автентикацията при този прототип е решено да се делегирана изцяло на външната услуга от Google Identity Services.

Голяма промяна има при ядрото на логиката за вземане на решение, което в случая на безсървърната архитектура, е разпределено между потребителския браузър и облачните функции на Firebase. Този подход позволява да се спестят ресурси на сървърната част, като част от операциите се изпълнят от React приложението. При този разработен прототип на СПВР е избрано към потребителския браузър да бъдат изнесени операциите по генериране на различните формуляри. Определянето на допустим модел, агрегирането и анализа на резултати се изпълняват като облачни функции. Това архитектурно решение е взето от съображения за сигурност, за да се избегне манипулация на JavaScript кода изпълняван в браузъра.

Ключов слой на безсървърната архитектура е **API Gateway**, представляващ единна точка за вход на всички клиентски заявки към системата. В него са интегрирани правилата за сигурност, даващи достъп до отделните функционалности. В конкретният прототип на СПВР, се използва облачното решение Google Cloud API Gateway, като

облачните функции са разработени на JavaScript и се изпълняват от Node.js. През API Gateway преминава и зареждането на front-end приложението от външната хостинг услуга, която в случая също се предоставя от Firebase.

В **слоят за база данни** се използва Firestore, която е документно ориентирана база данни част от облачните услуги Firebase. При Firestore данните се съхраняват не в таблици, а в йерархична структура от документи, представляващи двойки от ключ и стойност. Firestore също интегрира допълнително ниво на сигурност, изразяващо се в правила за достъп до данните, които се прилагат към облачните функции, опитващи да ги достъпят. Интерфейсът между базата данни и бизнес логиката е API предоставено от Firebase, което се ползва директно в кода на облачните функции.

Безсървърната архитектура използва асинхронна обработка на събитията, която се извършва от събитиен цикъл. Това позволява на системата да реагира на събития, като например клик на бутон, получени данни или изтичане на таймер без да блокира изпълнението на останалите задачи. Това осигурява плавна и ефективна работа на приложението. Събитийният цикъл се изпълнява на потребителският браузър, като логиката и обновяването на състоянията се управлява от библиотеката за управление на състояния Redux.

3.6. Сравнение на предложените софтуерни архитектури

При проектирането на прототип на СПВР архитектурния модел е ключов за постигането на баланс между ефективно използване на ресурси, мащабируемост, сигурност и удобство при разработка и поддръжка. Двете проектирани архитектури (трислойна и безсървърна) предлагат различни подходи към реализирането на бизнес логиката, изграждането на потребителския интерфейс и работата с данни.

Трислойната архитектура следва традиционния модел с ясно разграничени слоеве: потребителски интерфейс, бизнес логика и база данни. Този модел осигурява структурирано разделение на отговорностите, което улеснява поддръжката, тестването и разширяването на системата. В прототипа на СПВР, трислойната архитектура предоставя на сървъра за бизнес логика цялата обработка на данни, а клиентската страна е отговорна единствено за визуализацията и събирането им.

Предимствата на трислойната архитектура включват по-висока степен на контрол и централизация, което повишава сигурността, тъй като всички операции се извършват на сървъра. Синхронният характер на потока на събитията гарантира предвидимост на процесите и лесно отстраняване на грешки. Въпреки това, трислойната архитектура може да изисква по-големи хардуерни ресурси и усилия за управление на инфраструктурата, както и потенциално по-голямо забавяне при обработката поради нуждата от повече мрежова комуникация между слоевете.

Безсървърният модел акцентира върху използването на децентрализирани облачни функции и услуги, като така елиминира необходимостта от управлявани сървъри. В прототипа на СПВР бизнес логиката се изпълнява предимно в браузъра на клиента, а само някои изчислителни операции, които изискват сигурност се изпълняват в облака чрез Firebase Cloud Functions.

Предимства на безсървърната архитектура са много високата мащабируемост, тъй като облачните услуги автоматично се адаптират спрямо натоварването. Намаление на административните дейности по управление на инфраструктурата, което позволява по-бързо разработване и внедряване на нови функционалности. В същото време, разпределената природа на логиката изисква допълнителни мерки за сигурност, тъй като част от кода се изпълнява на клиентската страна. Това изисква внимателно проектиране и внедряване на повече проверки за авторизация в облачните функции.

Сравнение спрямо някои основни характеристики на проектираните трислойна и безсървърна архитектура на прототипите на СПВР, е направено таблично в Табл. 3.2.

В контекста на конкретните реализации на СПВР, трислойната архитектура предлага по-ясна обща структура и контрол над бизнес логиката. Тя е подходяща, ако системата изисква стабилна и предвидима среда с централизирано управление. Безсървърната архитектура е по-икономична от гледна точка на ресурси и по-адаптивна при вариращи натоварвания, като предлага по-гъвкав подход към поддръжката, който може да ускори внедряването.

Таблица 3.2. Сравнение между проектираните трислойна и безсървърна архитектура.

Характеристика	Трислойна архитектура	Безсървърна архитектура
Инфраструктура	Монолитна. Съсредоточена в един сървър	Разпределена между браузър и облачни функции
Разгръщане	Изисква поддръжка и управление на сървъри	Без необходимост от пряко управление на сървъри
Мащабируемост	Ограничена от ресурсите на сървърите	Автоматична спрямо натоварването
Сигурност	Има централизирано управление на достъпа	Прилага се към браузъра и облачните услуги
Производителност	По-бавна, заради нуждата от повече мрежова комуникация	По-бърза, заради по-малката нужда от мрежова комуникация
Разработка	По-ясна структура, по-свободна за разработка	По-сложна структура, с ограничения от облачните функции
Поддръжка	По-трудна, изисква администрация на сървъри	По-лесна, делегирана на облака
Обработване на събитията	Синхронно	Асинхронно

3.7. Изводи

Проектирането на две различни софтуерни архитектури за една и съща СПВР, има за цел да подпомогне избора на архитектурен подход. Този избор следва да се основава не само на функционалните изисквания, но и на нефункционалните характеристики като инфраструктурни възможности, технологичния опит на екипа и очакван мащаб на използване на системата.

Монолитната трислойна архитектура е логичният избор за институции, които разполагат със собствена сървърна инфраструктура и е достатъчна за академични цели като експерименти, обучение и демонстрации. От своя страна, безсървърната разпределена архитектура осигурява висока мащабируемост при динамични натоварвания. Тя е подходяща за организации, които разчитат на облачни услуги и трябва да обслужва голям брой потребители, например корпоративни или публични.

Преходът от монолитна трислойна към безсървърна архитектура, демонстрира как един експериментален академичен вариант на СПВР, може да еволюира до голяма корпоративна система със значима практическа приложимост.

ГЛАВА 4.
ЧИСЛЕНО ТЕСТВАНЕ НА МОДИФИЦИРАНИТЕ МЕТОДИ

В настоящата глава е описано извършеното числено тестване и анализ на предложените модификации върху алгоритъма за използване и моделите на метода на претеглената сума и метода на претегленото произведение. Описани са проведените експерименти използвайки:

- модификация на WSM и WPM, отчитаща компетентността на ЛРВ в различни области на компетентност, при задача за избор на платформа за електронна търговия;
- модификация на WSM, формализираща генериране на оценки за алтернативите, базирани на обективни данни и предпочитания на ЛВР, при задача за избор на монитор;
- модификация на WSM и WPM, формализираща генериране на теглови коефициенти за алтернативите, базирани на обективни данни и предпочитания на ЛВР, при задача за избор на софтуерна рамка за уеб разработка;
- обобщеният алгоритъм и математическите модели, комбиниращи модификациите;
- разработен прототип на система, подпомагаща вземането на решения.

Експериментите са проведени с три различни тестови задачи. За всяка от тях са формулирани отделни сценарии. С цел сравнимост между сценариите, те се реализират с едни и същи обективни входни данни и параметри на задачата.

4.1. Числено тестване на модификацията, отчитаща компетентността на ЛВР в различни области на компетентност

Числено тестване на предложената модификация, отчитаща различията в областите на компетентност на ЛВР, описана в Глава 2, т. 2.1, се изпълнява върху двата метода WSM и WPM. Тестването е реализирано за конкретна задача за избор на платформа за електронна търговия, като обещаващ инструмент за подобряване на бизнес процесите.

Изборът на задачата е мотивиран от актуалната протичащата дигитална трансформация и преживяната COVID-19 пандемия.

Проучванията подчертават промяна в нагласите на потребителите към електронната търговия след пандемията [Kawasaki и др. 2022]. Платформите за електронна търговия са основата, върху която бизнесът изгражда своето онлайн присъствие. С тяхна помощ отделите за търговия на дребно разширяват значително възможностите си чрез механизми за повишаване на продажбите [Niu и др., 2022]. Използването на такъв инструмент допринася за по-добра икономическа устойчивост, тъй като е възможно да се вземат предвид различни фактори, съобразени със специфичните нужди и изисквания на компаниите. В зависимост от типа клиенти, върху които бизнесът се фокусира, платформите могат да се разделят на такива, които са насочени към крайните потребители като бизнес към клиенти (B2C) или такива, които са насочени към бизнеса (B2B). Ценовите модели за ползване на платформата оказват значително влияние върху оперативните и финансови решения и имат ключова роля за рентабилността на бизнеса [Chang и др., 2022]. Други основни съображения по време на процеса по оценка и подбор са въпросите за внедряването и информационната сигурност при плащанията [Hassan и др., 2020]. Освен това, важен фактор е интегрирането в цялостната екосистемата от платформи подпомагащи търговията, като системи за управление на съдържанието, счетоводството и връзката с клиентите [Engert и др., 2022].

Изборът на платформа за електронна търговия е комплексна задача. Платформата трябва да отговаря на критерии от различни области на компетентност. Това прави задачата подходяща да бъде решена с предложената модификация, отчитаща компетентността на ЛВР в различни области на компетентност, в условията на групово вземане на решение.

4.1.1. Сценарии за тестване

За да се изследва спецификата на модификацията, отчитаща степента на компетентност на ЛВР са изследвани два сценария за метода на претеглената сума, използвайки разработения прототип. При първият сценарии (Сценарий 1) се формулира и решава задача на база класическия модел на WSM (1.3) – (1.5). Вторият (Сценарий 2) се реализира чрез формулиране и решаване на задача на база на предложената

модификация, в съответствие с (2.7) за WSM и (2.11) за WPM. Модификацията на метода на претегленото произведение е тествана и сравнена с модификацията на метода на претеглената сума.

4.1.2. Формулиране на задачата за числено тестване

Конкретната задача за избор на платформа за електронна търговия се отнася до малка компания, която предварително е задала три подходящи според нейните цели B2C платформи, а именно: WooComerce, разглеждана като алтернатива **A1**; Shopify-Basic, разглеждана като алтернатива **A2**; и BigComerce Essentials-Standard, разглеждана като алтернатива **A3**. По време на процеса по идентифициране на критериите за оценка е необходимо да се има предвид, че управлението на бизнес с електронна търговия е комплексна инициатива. Това означава, че за успешното му развитие е необходимо да се разчита на експерти с умения от различни области – продажби, маркетинг, информационни технологии, бизнес организация и управление.

Отчитайки и необходимостта от подпомагане на процеса на администриране при групово вземане на решение, се предлага критериите за оценка да се разделят на групи. Съгласно **Стъпка 3** от предложения алгоритъм, описан в точка 2.1.1, за конкретната задача са идентифицирани четири групи от критерии. Използваните критерии, са определени на база на публично достъпна информация, предоставена от официалните сайтове на платформите [https://woocommerce.com/; https://www.shopify.com/; https://www.bigcommerce.com/]. В Таблица 4.1. са показани както групите, така и съответните критерии, формиращи всяка група.

Таблица 4.1. Критерии за оценка на алтернативи, групирани по област на компетентност

Група 1: Общи критерии	
C1	Ценови модел
C2	Опции за клиентски разплащания
C3	Брой административни потребители (касиер, складов оператор)
C4	Канали за техническа поддръжка
C5	Брой безплатни шаблони за уеб интерфейс
Група 2: Търговски критерии	
C6	Управление на инвентара
C7	Опростен процес за пазаруване
C8	Управление на доставките
C9	Инструмент за търговски анализи
C10	Управление на отстъпки и промоционални кампании
Група 3: Технически критерии	
C11	Внедряване и поддръжка (DevOps)
C12	Сигурност на информацията и плащанията
C13	Адаптивен дизайн
C14	Интеграции с други приложения
C15	Скорост и производителност
C16	Оптимизация за търсещи машини (SEO)
Група 4: Маркетингови критерии	
C17	Персонализация на потребителския интерфейс (UX/UI)
C18	Управление на потребителските отзиви
C19	Канали за реклама

Група 1 включва общи критерии, които са важни за функционирането на бизнеса и използването на платформата за електронна търговия като цяло. Важно е платформата да отговаря на целите, възможностите и мащаба на бизнеса като цена, персонал и планове за бъдещо развитие. От друга гледна точка, това може да е група, която обединява критерии, за които не може да се определи принадлежността им към останалите групи. Това е идея, за това как може да се постъпи в такива ситуации.

Група 2 е свързана с търговията и организацията на търговския процес. Включени са критерии като управление на инвентара, който е важен за организацията

на продажбите или инструмент за анализ на продажбите, което допринася за подобряване на планирането.

Група 3 обхваща критерии, свързани с техническите характеристики на платформата. Тъй като работната среда е онлайн, платформата за електронна търговия трябва да отговаря на изискванията за скорост и сигурност на плащанията. Поддръжката, интеграцията с други приложения и способността да бъдат открити от търсачките също са важни критерии в тази група.

Критериите, формиращи **Група 4**, са свързани с рекламата. Важно е платформата за електронна търговия да предлага персонализация на шаблоните за потребителски интерфейс. Вида на електронния магазин може да привлича клиенти и да осигурява удобство, лекота за използване и приятно изживяване за клиентите. Друга важна функционалност, която се предлага на бизнеса е възможността да достигне до целевата си група клиенти чрез различни канали за реклама, като електронна поща и социални мрежи.

Определена е група от експерти, чиито компетенции напълно покриват така формираните групи критерии. Същевременно се определят различни теглови коефициенти по начин, който да отразява индивидуалните компетенции на отделното ЛВР. Определените теглови коефициенти за компетентност на ЛВР, съгласно ограничение (2.3), при Сценарий 2 (тестващ модифицираните методи WSM и WPM) са изложени в Таблица 4.2.

Таблица 4.2. Теглови коефициенти за компетентност на ЛВР, спрямо групите критерии.

Групи критерии	**Теглови коефициенти за компетентност на ЛВР**			
	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-4
Група 1 (Общи критерии)	0.25	0.25	0.25	0.25
Група 2 (Търговски критерии)	0.15	0.50	0.15	0.20
Група 3 (Технически критерии)	0.14	0.14	0.57	0.15
Група 4 (Маркетингови критерии)	0.25	0.25	0.10	0.40

Определените експерти за формиране на груповото решение са както следва: (1) **ЛВР-1** – собственик на бизнеса – от него се очаква да има визия за развитието и управлението на бизнеса, както и за глобалните цели, които се преследват; (2)

Мениджър продажби (**ЛВР-2**) лице, упълномощено да организира и управлява продажбите. Неговата специалност е в областта на анализа върху търсенето и предлагането, както и в ценообразуването. Основателно, той получава най-голяма относителна тежест върху критериите от Група 2. (3) Техническо лице или софтуерен инженер (**ЛВР-3**), който има отношение към експлоатацията и техническата поддръжка на платформата. Той отговаря с квалификацията си за критериите от Група 3. (4) Маркетингов мениджър (**ЛВР-4**) – фокусът на този експерт е върху потребителското изживяване и възможностите за комуникация и взаимодействие с клиентите. Друга важна характеристика от работата му е привличането на нови клиенти и той получава най-висока тежест върху критериите от Група 4. Важно е да се отбележи, че за Група 1, която съдържа общите критерии, ЛВР допринасят за крайното решение с еднаква важност (вж. Таблица 4.2).

Отчитайки определените по-горе входни данни за задачата за избор на платформа за електронна търговия, е възможно да се генерират форми, с помощта на които се дефинира задачата за групово вземане на решение. За всяко ЛВР се генерира матрица за индивидуално вземане на решение, чрез която съответния експерт определя тегловите коефициента за важност на критериите и респективно дава оценки за представянето на всяка от алтернативите спрямо определените критерии. Важно е да се отбележи, че ЛВР могат да използват стойности за оценки и коефициенти в диапазона между 0 и 1, като по-голямата стойност показва по-добро представяне.

Съгласно предложената структура в Таблица 2.2 (вж. Глава 2, т. 2.1.2), входните данни за конкретната задача са представени в матрицата за групово вземане на решение, както е показано в Таблица 4.3:

Таблица 4.3. Матрица за групово вземане на решение с отчитане компетентността на ЛВР.

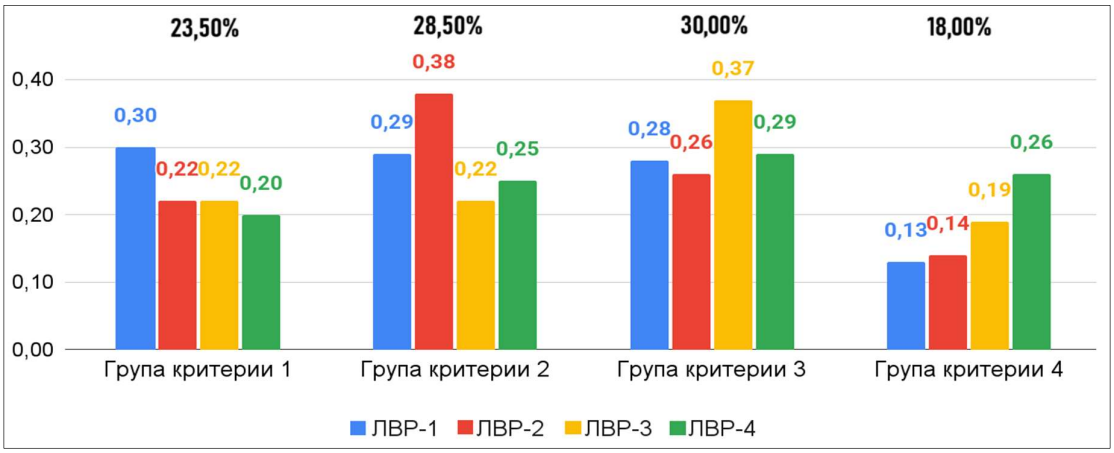
Критерии						Тегла за компетентност на ЛВР				Оценки на алтернативите											
Групи	№	Тегла за важност на критериите								**A1** (WooComerce)				**A2** (Shopify)				**A3** (BigComerce)			
		ЛВР1	ЛВР2	ЛВР3	ЛВР4	ЛВР1	ЛВР2	ЛВР3	ЛВР4	ЛВР1	ЛВР2	ЛВР3	ЛВР4	ЛВР1	ЛВР2	ЛВР3	ЛВР4	ЛВР1	ЛВР2	ЛВР3	ЛВР4
Група1	C1	0.10	0.03	0.02	0.01	0.25	0.25	0.25	0.25	0.43	0.38	0.62	0.57	0.24	0.62	0.52	0.57	0.24	0.62	0.52	0.57
	C2	0.06	0.08	0.02	0.07					0.14	0.33	0.05	0.29	0.14	0.33	0.05	0.33	0.24	0.14	0.33	0.10
	C3	0.05	0.02	0.05	0.01					0.10	0.10	0.10	0.19	0.29	0.14	0.10	0.19	0.33	0.29	0.29	0.24
	C4	0.07	0.04	0.10	0.04					0.40	0.14	0.50	0.33	0.81	0.50	0.57	0.62	0.81	0.50	0.57	0.62
	C5	0.02	0.05	0.03	0.07					0.24	0.19	0.19	0.24	0.24	0.33	0.19	0.29	0.29	0.05	0.24	0.33
Група2	C6	0.08	0.10	0.07	0.04	0.15	0.50	0.15	0.20	0.39	0.45	0.47	0.44	0.35	0.88	0.51	0.53	0.75	0.35	0.69	0.77
	C7	0.03	0.07	0.04	0.06					0.50	0.51	0.49	0.48	0.62	0.58	0.49	0.53	0.49	0.42	0.51	0.55
	C8	0.05	0.08	0.02	0.03					0.41	0.33	0.44	0.39	0.52	0.67	0.55	0.61	0.48	0.40	0.49	0.52
	C9	0.10	0.06	0.06	0.06					0.66	0.71	0.59	0.61	0.69	0.89	0.66	0.71	0.60	0.71	0.59	0.61
	C10	0.03	0.07	0.03	0.06					0.35	0.42	0.47	0.38	0.55	0.72	0.71	0.62	0.51	0.62	0.55	0.49
Група3	C11	0.08	0.03	0.09	0.02	0.14	0.14	0.57	0.15	0.56	0.52	0.95	0.40	0.61	0.40	0.30	0.45	0.59	0.39	0.30	0.44
	C12	0.07	0.08	0.06	0.04					0.61	0.69	0.71	0.59	0.48	0.59	0.60	0.49	0.51	0.55	0.59	0.41
	C13	0.04	0.03	0.06	0.06					0.31	0.40	0.65	0.25	0.42	0.50	0.59	0.62	0.36	0.55	0.57	0.45
	C14	0.03	0.05	0.07	0.04					0.69	0.72	0.85	0.74	0.33	0.42	0.33	0.50	0.45	0.46	0.40	0.42
	C15	0.03	0.04	0.04	0.05					0.35	0.40	0.62	0.33	0.55	0.61	0.50	0.46	0.51	0.60	0.49	0.44
	C16	0.03	0.03	0.05	0.08					0.51	0.46	0.63	0.44	0.53	0.57	0.49	0.60	0.49	0.50	0.38	0.50
Група4	C17	0.03	0.06	0.09	0.09	0.25	0.25	0.10	0.40	0.32	0.37	0.41	0.39	0.47	0.42	0.39	0.39	0.51	0.41	0.42	0.44
	C18	0.06	0.04	0.05	0.07					0.66	0.63	0.72	0.82	0.67	0.72	0.74	0.81	0.76	0.77	0.67	0.79
	C19	0.04	0.04	0.05	0.10					0.61	0.62	0.71	0.62	0.34	0.45	0.55	0.49	0.77	0.82	0.88	0.85

Към така формираната матрица за групово вземане на решение при избор на платформа за електронна търговия, за тестване на модификацията е необходимо да се използва един от двата предложени математически модели (2.7) или (2.11) за агрегиране на оценките на алтернативите. Анализът на получените резултати от проведените експерименти са описани в следващия раздел.

4.1.3. Анализ на входните данни

На база на данните от Таблици 4.3 – 3 алтернативи, 4 ЛВР, 19 критерия, разпределени в 4 групи, е направен анализ на входните данни, преди да се приложат математическите модели на метода на претегленото произведение и двата сценария за метода на претеглената сума.

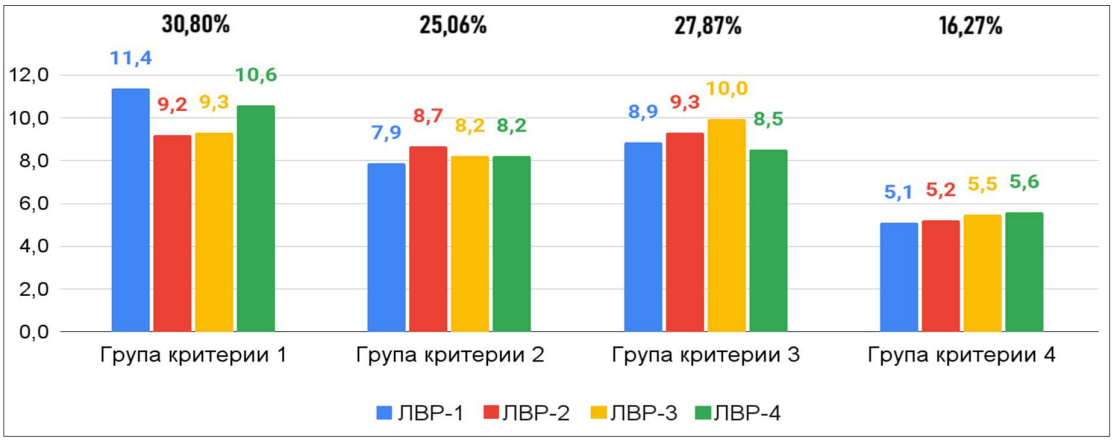
Всяко ЛВР е посочило, че най-важни за него са критериите в групата, където е с най-висока компетентност, това се вижда от Фиг. 4.1.



Фиг. 4.1. Съотношения на определени от ЛВР, коефициенти за важност на критериите, спрямо групите критерии и процентното съотношение между групите критерии.

Както се очаква, Група 3, която е с най-много критерии (6 бр.) има и най-висока тежест (30%), а Група 4 с най-малко критерии (3 бр.) – най-ниска (18%). Интересното е, че Групи 1 и 2 съдържат равен брой критерии (по 5 бр.), но Група 2 е оценена значително по-високо (28,5% спрямо 23,5%), което показва разлика във възприетата им значимост при избора на платформа за електронна търговия.

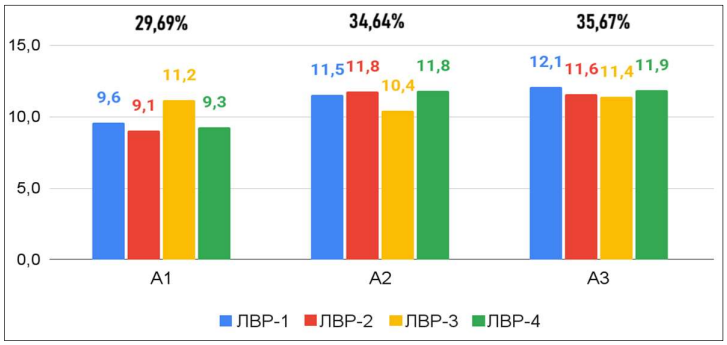
Предпочитанията към алтернативите, изразени единствено чрез оценки, като се изключи влиянието на коефициентите за важност и компетентност са представени на Фиг. 4.2.



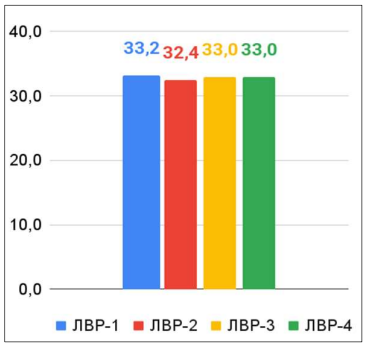
Фиг. 4.2. Съотношения на определените от ЛВР оценки за алтернативите, спрямо групите критерии и процентното съотношение между групите критерии.

Както при коефициентите за важност на критериите, така и при оценките за алтернативите, отново се потвърждава тенденцията ЛВР да оценяват най-високо в групата, където са най-компетентни. Очаквано, поради малкия брой критерии, по Група 4 се събират най-малко оценъчни точки (16,27%), но е интересно да се отбележи, че ЛВР са оценили най-високо алтернативите по Група 1 с дял от 30,80%.

Същите данни, разгледани от гледна точка на алтернативите, показват превес на оценките в полза на А2 и А3, докато А1 получава значително по-ниски стойности, както се илюстрира на Фиг. 4.3.а. Общият брой оценки, дадени от всяко лице, е представен на Фиг.4.3.б.



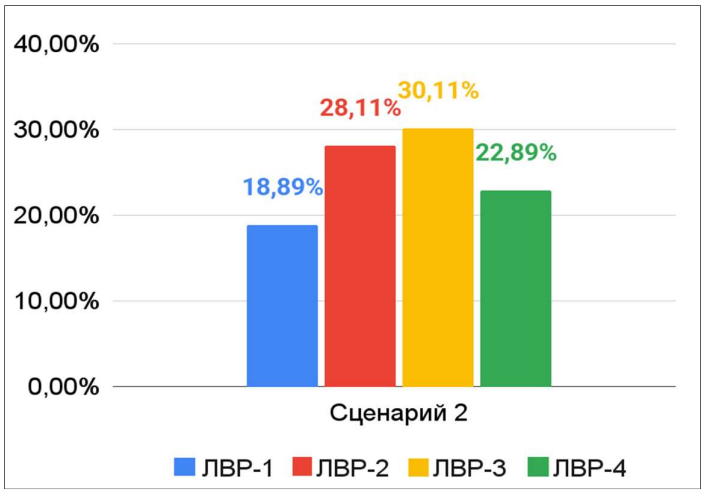
а) Спрямо алтернативите



б) Спрямо ЛВР

Фиг. 4.3. Съотношения на определените от ЛВР оценки на алтернативите.

Интересно е да се покаже, какво би било добавеното влияние на отделните ЛВР при формиране на крайното решение, използвайки предложената модификация. Вземайки предвид коефициентите за компетентност на ЛВР за групите критерии и броят на критериите в групите се установява, че влиянието на ЛВР-3 се увеличава до 30,11%, както и за ЛВР-2 до 28,11%, за сметка на ЛВР-4 (22,89%) и ЛВР-1 (18,89%). Най-ниският резултатът за ЛВР-1 е очакван тъй като той не е изявен в нито една група критерии, както може да се види на Фиг. 4.4.

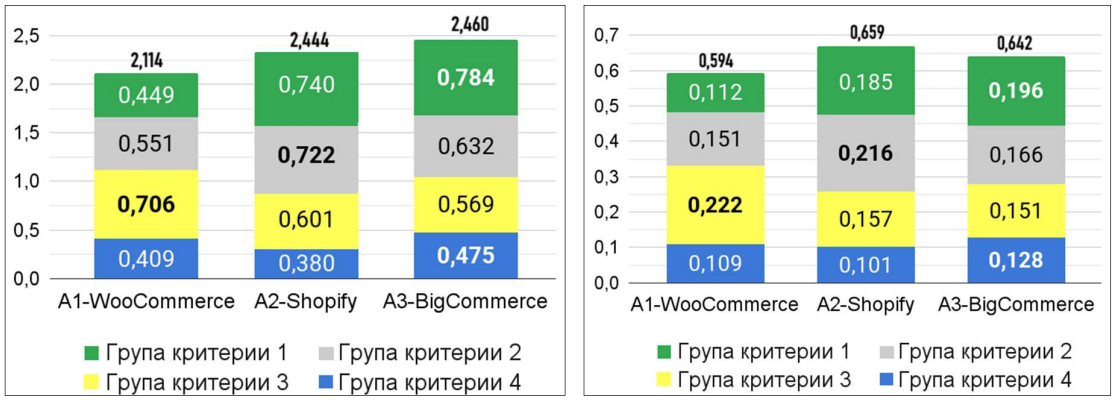


Фиг. 4.4. Процентно съотношение на коефициентите за компетентност на ЛВР, обобщено за всички критерии.

Този предварителен анализ на входните данни е ключов за интерпретацията на резултатите получени след прилагане на математическите модели на методите за групово вземане на решение.

4.1.4. Резултати от численото тестване на модификацията на метода на претеглената сума

При Сценарий 1 се използва класическия метод на претеглената сума (1.3) – (1.5). При Сценарий 2, върху същите данни и при условие, че се групират критериите (с цел подпомагане на администрирането на задачата), се използва модифицирания метод на претеглената сума (2.7) – (2.10). Получените резултати от груповото решение класират алтернативите при двата сценария, както е илюстрирано на Фиг. 4.5 (а, б).



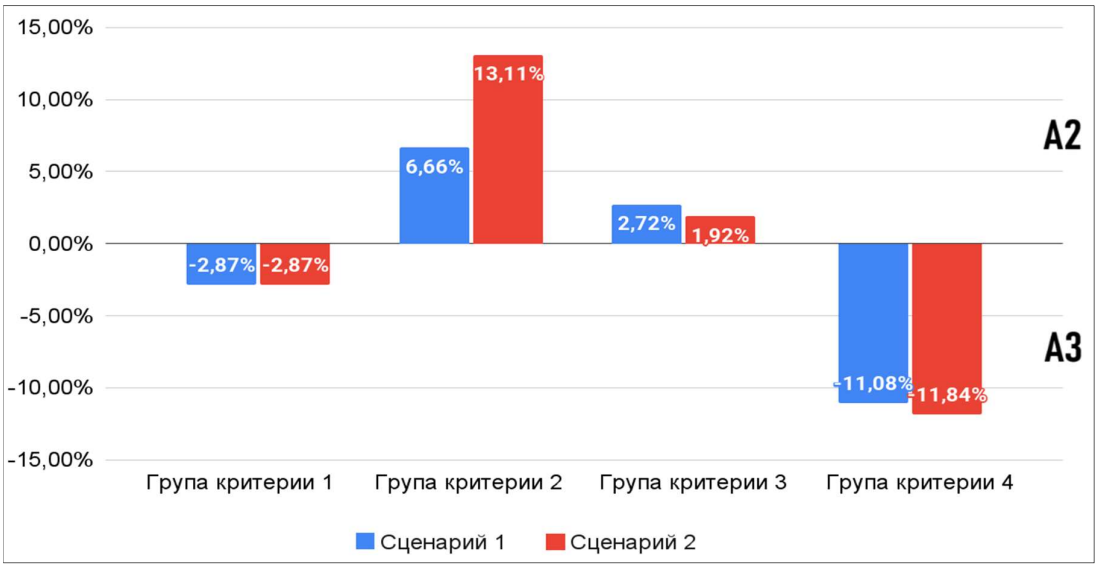
а) Сценарий 1 б) Сценарий 2

Фиг. 4.5. Класиране на платформите за електронна търговия при Сценарий 1 и Сценарий 2.

При Сценарий 1, реализиращ, класическия WSM, алтернативата A3-BigCommerce е класирана на първо място с общ резултат от 2,460, следвана от A2-Shopify (2,444) и A1-WooCommerce (2,114).

При Сценарий 2, реализиращ, модифицирания WSM, най-предпочитаната алтернатива е A2-Shopify с общ резултат от 0,659. На второ място след нея е алтернативата A3-BigCommerce (0,642) и последна в класацията е алтернативата A1-WooCommerce (0,594). За разлика от Сценарий-1, тук агрегираните стойности на алтернативите са приблизително четири пъти по-ниски. Тази разлика в мащабите се дължи на ефекта от въведените допълнителни коефициенти за компетентност, които разделят влиянието между всички ЛВР (вж. Фиг. 4.4.).

Резултатите по групи, представени на Фиг. 4.5.б, показват промяна в съотношението на приноса на отделните групи критерии при формирането на обобщените оценки. Например, при Сценарий 2 се наблюдава увеличение в относителния дял на Група 2 с 0,216 агрегирани точки за сметка на намаление в приноса на Група 1 до 0,185 агрегирани точки. Още при анализа на съотношенията между оценките, дадени от ЛВР спрямо алтернативите (вж. Фиг. 4.3), се очертава конкуренция между алтернативи A2 и A3. Тъй като в Сценарий 2 A2 изпреварва A3 в крайното класиране, последващият анализ на пропорциите между групите критерии се фокусира върху сравнение между тези две алтернативи, както е показано на Фиг. 4.6.



Фиг. 4.6. Изменение на съотношенията на агрегираните оценки по групи критерии за алтернатива A2-Shopify спрямо алтернатива A3-BigCommerce при двата сценария.

Ключовата причина за разместването в класирането на алтернативите се забелязва в Група критерии 2, където тежестта, която се придава на мнението на търговския мениджър (ЛВР-2) е най-висока (0,50 от 1,00 – вж. Таблица 4.2). Предварителният анализ на входните данни от Таблица 4.3 показва, че всички оценки на ЛВР-2 за алтернативите по критериите в тази група са в полза на А2-Shopify (общо 3,740). Допълнително, от Фиг. 4.1. може да се установи, че той е лицето направило най-високото разпределение на коефициентите за важност на критериите, съставящи Група 2 (0,38 от 1,00). Следователно тази комбинация е довела до повишаване на съотношението между оценките на алтернативи А2 и А3 по критериите в Група 2 от 6,66% (при Сценарий 1) до 13,11% (при Сценарий 2). Тази промяна се оказва достатъчна, за да се получи различно класиране на алтернативите.

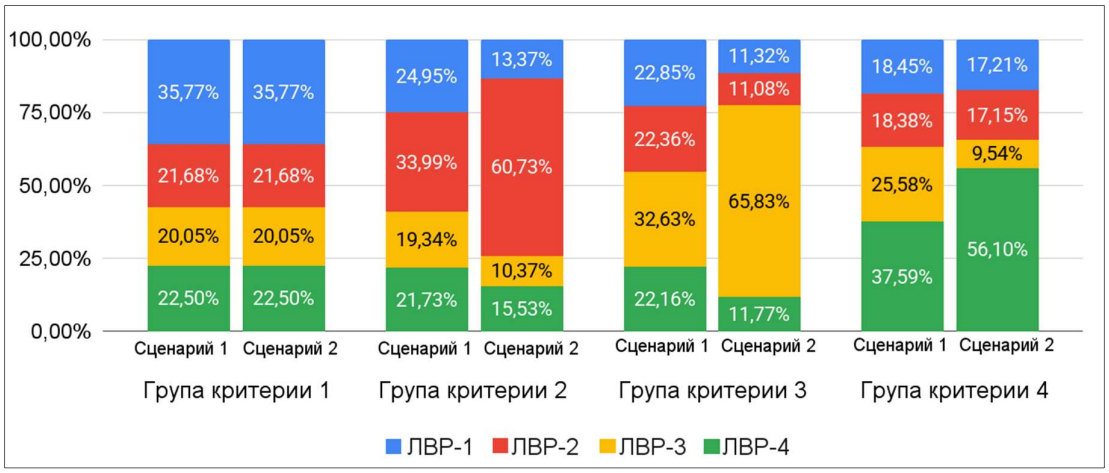
Използвайки разработения прототип на система за подпомагане вземането на решения и данните от матрицата за вземане на решение от Таблица 4.3 е генерирана таблица, чрез която може да се направи сравнителен анализ на двата сценария. Един изглед на част от модула за анализ на резултатите е представен на Фиг. 4.7.

Целта на този анализ е детайлно открояване на влиянието на нововъведените теглови коефициенти за компетентност на ЛВР и улеснено сравнение между двата сценария. Участието на отделните ЛВР, в агрегираните по групи критерии оценките за алтернативите, е изразено като процентно отношение.

Теглови коефициенти за компетентност на ЛВР				A1 WooCommerce				A2 Shopify - Basic				A3 BigCommerce essentials - Standard			
ЛВР-1 (Business Owner)	ЛВР-2 (Sales Manager)	ЛВР-3 (Software Engineer)	ЛВР-4 (Marketing Manager)	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-4	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-4	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-4
Група 1: Общи критерии															
0,25	0,25	0,25	0,25												
Сценарий 1 - участие на ЛВР				41,66%	19,82%	16,37%	22,15%	35,90%	21,73%	19,34%	23,03%	32,28%	22,71%	22,82%	22,20%
Сценарий 2 - участие на ЛВР				41,66%	19,82%	16,37%	22,15%	35,90%	21,73%	19,34%	23,03%	32,28%	22,71%	22,82%	22,20%
Сценарий 1 - агрегирана оценка на алтернативата				0,449				0,740				0,784			
Сценарий 2 - агрегирана оценка на алтернативата				0,112				0,185				0,196			
Група 2: Търговски критерии															
0,15	0,50	0,15	0,20												
Сценарий 1 - участие на ЛВР				26,01%	32,53%	20,12%	21,34%	21,89%	39,59%	17,61%	20,92%	27,52%	28,85%	20,63%	23,00%
Сценарий 2 - участие на ЛВР				14,21%	59,25%	11,00%	15,55%	10,98%	66,20%	8,83%	13,99%	15,73%	54,96%	11,79%	17,52%
Сценарий 1 - агрегирана оценка на алтернативата				0,551				0,722				0,632			
Сценарий 2 - агрегирана оценка на алтернативата				0,151				0,216				0,166			
Група 3: Технически критерии															
0,14	0,14	0,57	0,15												
Сценарий 1 - участие на ЛВР				20,74%	21,05%	40,08%	18,12%	23,54%	22,75%	27,62%	26,09%	24,74%	23,58%	28,67%	23,01%
Сценарий 2 - участие на ЛВР				9,24%	9,38%	72,72%	8,65%	12,61%	12,18%	60,23%	14,97%	13,04%	12,43%	61,53%	13,00%
Сценарий 1 - агрегирана оценка на алтернативата				0,706				0,601				0,569			
Сценарий 2 - агрегирана оценка на алтернативата				0,222				0,157				0,151			
Група 4: Маркетингови критерии															
0,25	0,25	0,10	0,40												
Сценарий 1 - участие на ЛВР				18,01%	17,67%	26,52%	37,80%	17,85%	18,93%	26,19%	37,02%	19,30%	18,56%	24,27%	37,87%
Сценарий 2 - участие на ЛВР				16,87%	16,55%	9,94%	56,65%	16,76%	17,78%	9,84%	55,62%	17,85%	17,16%	8,98%	56,02%
Сценарий 1 - агрегирана оценка на алтернативата				0,409				0,380				0,475			
Сценарий 2 - агрегирана оценка на алтернативата				0,109				0,101				0,128			
Сценарий 1 - агрегирани оценки за отделните ЛВР %				26,02%	23,13%	27,23%	23,62%	25,91%	26,82%	21,93%	25,33%	26,81%	23,69%	23,89%	25,62%
Сценарий 2 - агрегирани оценки за отделните ЛВР %				18,02%	25,35%	34,85%	21,77%	19,25%	33,42%	24,18%	23,15%	20,57%	27,52%	26,32%	25,59%
Сценарий 1 - класиране на алтернативите				2,114				2,444				2,460			
Сценарий 2 - класиране на алтернативите				0,594				0,659				0,642			
19%	28%	30%	23%	Средно аритметично влияние на тегловите коефициенти за компетентност на ЛВР за всички критерии											

Фиг. 4.7. Примерен изглед от модула за линеен анализ на резултатите от прототип на система за подпомагане вземането на решения.

В групата на общите критерии, коефициентите за компетентност на ЛВР са еднакви. Това води до еднакво процентното съотношение на участието в агрегираните оценки от всеки ЛВР при двата сценария, както може да се види и на Фиг. 4.7. Промяната се изразява единствено в четирикратното намаление на стойността на агрегираните оценки. Изводът е, че при равни коефициенти за компетентност на ЛВР, класическия и модифицирания метод на претеглената сума са еквивалентни. Това се илюстрира с Група критерии 1 от следната Фиг. 4.8.



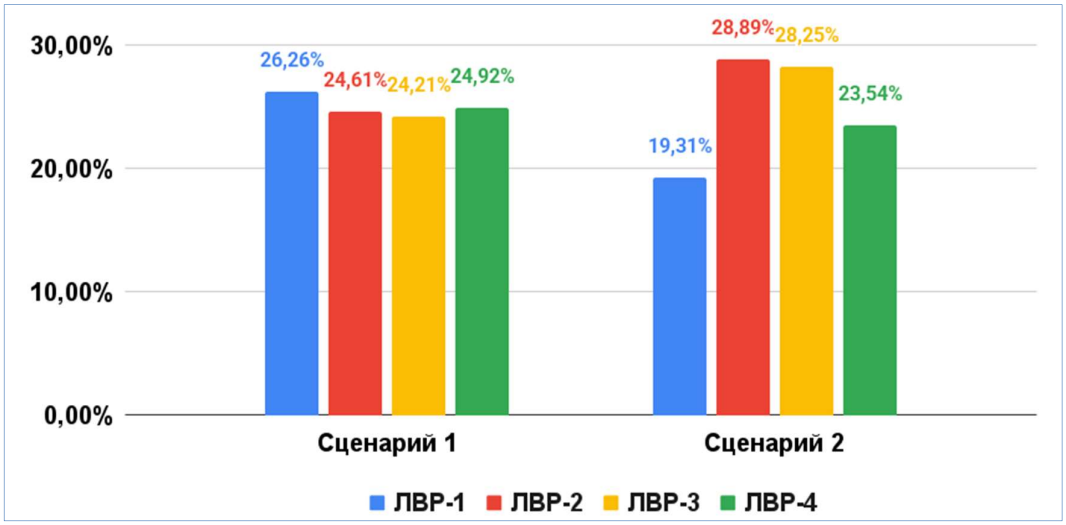
Фиг. 4.8. Сравнение на влиянието на ЛВР върху агрегираните оценки за всички алтернативи, по групи критерии, при двата сценария.

При Група критерии 2 се забелязва разместване на участието на ЛВР-1 и ЛВР-4 в агрегирането на оценките. При Сценарий 1, в по-изгодна позиция е собственика на бизнеса с 24,95% спрямо маркетинговия мениджър с 21,73%. При Сценарий 2, по-високият коефициент за компетентност от 0,20 за ЛВР-4, придава на мнението му по-голяма тежест като получава 15,53% в сравнение с ЛВР-1, който има коефициент от 0,15 и пресметнато участие от 13,37%. Пропорциите във влиянието между ЛВР-1 и ЛВР-3 се запазват, тъй като имат равни коефициенти от 0,15.

Определянето на коефициентите за компетентност при техническите критерии е с един водещ експерт и трима с приравнено влияние. При това положение почти не се променят пропорциите на влияние върху агрегираните оценки. От Фиг. 4.7 се забелязва, че изключение прави случаят на А3, където ЛВР-4 изпреварва ЛВР-2 с 0,57%, което се дължи на малко по-високият му коефициент за компетентност от 0,15 спрямо 0,14 за ЛВР-2.

За критериите от Група 4, касаеща маркетингови характеристики, е определен един водещ експерт – ЛВР-4 (0,40) и един, който има значително по-нисък коефициент за компетентност – ЛВР-3 (0,10), спрямо останалите участници. Това води до увеличаване на влиянието на маркетинговият мениджър и намаляване на тежестта на софтуерния инженер. Трябва да се отбележи, че при Сценарий 1 мнението на софтуерния инженер е второ по важност при всички алтернативи. При Сценарий 2 обаче, търговският и маркетинговият мениджър, имат коефициенти на компетентност от 0,25, които изпреварват софтуерния инженер и неговото влияние в агрегираните оценки на всички алтернативи е най-ниско.

Влиянието на отделните ЛВР при формирането на окончателното групово решение при двата сценария е представено чрез диаграмата от Фиг. 4.9.

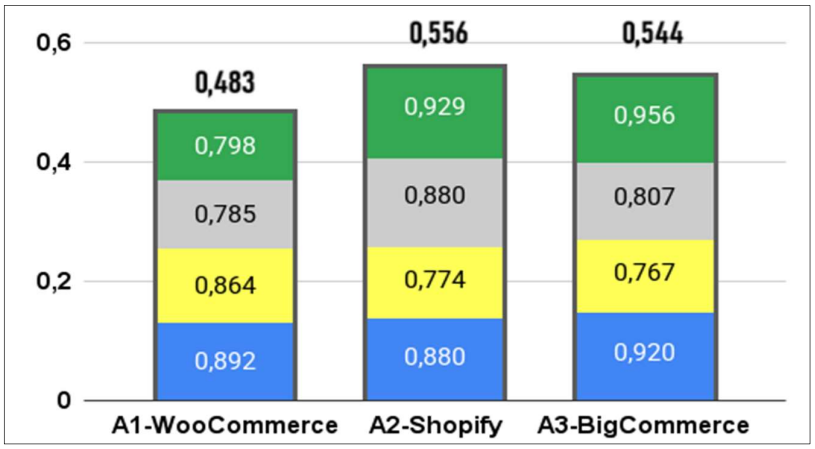


Фиг. 4.9. Сравнение на влиянието на отделните ЛВР при формирането на окончателното групово решение, между класическия и модифицирания метод на претеглената сума.

При Сценарий 2, който се реализира на база на предложения модифициран модел, се забелязва балансиране на финалното влияние на отделните ЛВР. Това е в контраст с влиянието по групи критерии, където например при Група 3 водещият експерт има влияние, надвишаващо шесткратно това на останалите участници (вж. Фиг. 4.8). Установеното балансиране се дължи на определянето на тегловите коефициенти за компетентност по начин, при който всяка от групите критерии има по един водещ експерт.

4.1.5. Резултати от численото тестване на модификацията на метода на претегленото произведение

Модификацията на метода на претегленото произведение се тества върху входните данни дефинирани в Таблица 4.2 и Таблица 4.3. Предварителният анализ направен върху трите атрибута: (1) коефициенти за важност на критериите, (2) оценки за алтернативите и (3) коефициенти за компетентност на ЛВР илюстриран на Фиг. 4.1 – Фиг. 4.4 е валиден и в този случай. Получените резултати от груповото решение класират алтернативите както е илюстрирано на Фиг. 4.10.



Фиг. 4.10. Класиране на платформите за електронна търговия при използване на модифицирания модел на претегленото произведение.

Може да се отбележи, че се запазва класирането получено от модифицирания WSM. Има разлики в числените стойности на агрегираните оценки, както следва: A2-Shopify (0,556), A3-BigCommerce (0,544) и A1-WooCommerce (0,483).

За да се направи сравнение на ефекта от нововъведените коефициенти за компетентност върху влиянието на ЛВР при WSM и WPM се извършва линеен анализ на агрегираните резултати. За тази цел е необходимо прилагане на логаритмични преобразувания, използвайки техниката на логаритмична линеаризация (1.17) върху резултатите от метода на претегленото произведение. Тези операции са изпълнени чрез модула за линеен анализ на прототипа на системата за подпомагане вземането на решения. Изглед от сравнението на резултатите е показан на Фиг. 4.11.

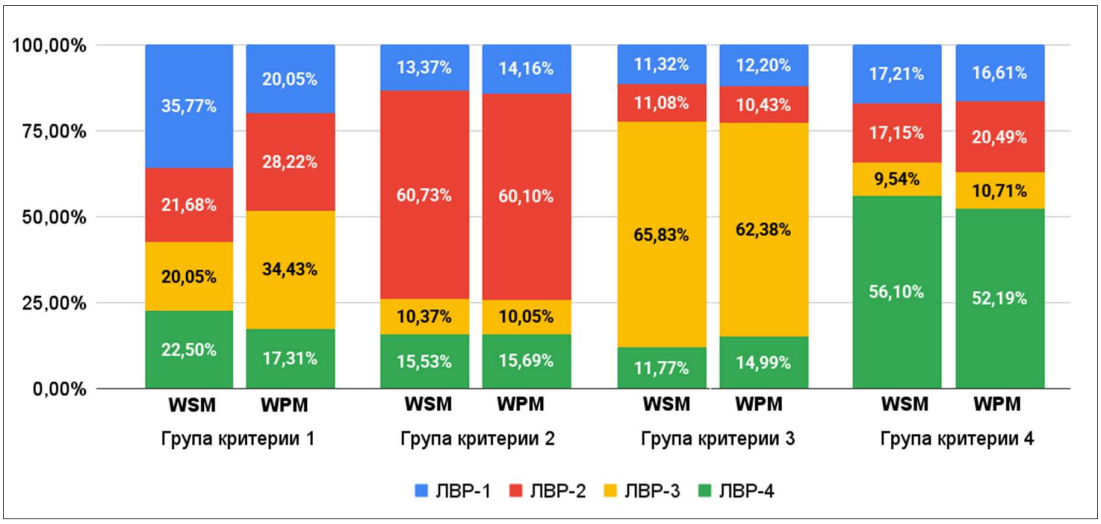
Както се очаква, доминацията на едно ЛВР в групата, където има най-висок коефициент за компетентност се запазва. Все пак по групи критерии се наблюдават разлики в процентното влияние на ЛВР, поради мултипликативната природа на метода на претегленото произведение и неговата чувствителност към крайни стойности. Например, при Група 2 и алтернатива A3, влиянието на ЛВР-2 се увеличава от 54,96% (при WSM) до 68,28% (при WPM). Обратен ефект се отчита при Група 3 и алтернатива A1, където влиянието на ЛВР-3 намалява от 72,72% на 40,81%. Подобна промяна се наблюдава и при Група 1, въпреки че там коефициентите за компетентност са равни. Това показва, че разликите не се дължат на тегловите коефициенти, а на спецификите на използвания метод.

Теглови коефициенти за компетентност на ЛВР				A1 WooCommerce				A2 Shopify - Basic				A3 BigCommerce essentials - Standard			
ЛВР-1 (Business Owner)	ЛВР-2 (Sales Manager)	ЛВР-3 (Software Engineer)	ЛВР-4 (Marketing Manager)	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-4	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-4	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-4
Група 1: Общи критерии															
0,25	0,25	0,25	0,25												
WSM - участие на ЛВР				41,66%	19,82%	16,37%	22,15%	35,90%	21,73%	19,34%	23,03%	32,28%	22,71%	22,82%	22,20%
WPM - участие на ЛВР				20,39%	28,06%	32,25%	19,30%	12,90%	29,43%	45,81%	11,86%	30,01%	27,03%	26,72%	16,24%
WSM - агрегирана оценка на алтернативата				0,112				0,185				0,196			
WPM - агрегирана оценка на алтернативата				0,798				0,929				0,956			
Група 2: Търговски критерии															
0,15	0,50	0,15	0,20												
WSM - участие на ЛВР				14,21%	59,25%	11,00%	15,55%	10,98%	66,20%	8,83%	13,99%	15,73%	54,96%	11,79%	17,52%
WPM - участие на ЛВР				13,25%	61,37%	9,43%	15,94%	21,75%	44,01%	14,36%	19,88%	10,65%	68,28%	8,16%	12,90%
WSM - агрегирана оценка на алтернативата				0,151				0,216				0,166			
WPM - агрегирана оценка на алтернативата				0,785				0,880				0,807			
Група 3: Технически критерии															
0,14	0,14	0,57	0,15												
WSM - участие на ЛВР				9,24%	9,38%	72,72%	8,65%	12,61%	12,18%	60,23%	14,97%	13,04%	12,43%	61,53%	13,00%
WPM - участие на ЛВР				18,27%	14,67%	40,81%	26,25%	10,71%	9,32%	69,39%	10,58%	10,31%	9,17%	67,48%	13,04%
WSM - агрегирана оценка на алтернативата				0,222				0,157				0,151			
WPM - агрегирана оценка на алтернативата				0,864				0,774				0,767			
Група 4: Маркетингови критерии															
0,25	0,25	0,10	0,40												
WSM - участие на ЛВР				16,87%	16,55%	9,94%	56,65%	16,76%	17,78%	9,84%	55,62%	17,85%	17,16%	8,98%	56,02%
WPM - участие на ЛВР				17,30%	21,33%	9,98%	51,39%	17,54%	18,96%	10,13%	53,37%	14,22%	21,69%	12,61%	51,48%
WSM - агрегирана оценка на алтернативата				0,109				0,101				0,128			
WPM - агрегирана оценка на алтернативата				0,892				0,880				0,920			
WSM - участие на ЛВР				18,02%	25,35%	34,85%	21,77%	19,25%	33,42%	24,18%	23,15%	20,57%	27,52%	26,32%	25,59%
WPM - участие на ЛВР				17,11%	35,39%	22,90%	24,60%	14,89%	21,56%	41,43%	22,12%	12,43%	33,04%	36,07%	18,46%
WSM - агрегирана оценка на алтернативата				0,594				0,659				0,642			
WPM - агрегирана оценка на алтернативата				0,483				0,556				0,544			
19%	28%	30%	23%	Средно аритметично влияние на тегловите коефициенти за компетентност на ЛВР за всички критерии											

Фиг. 4.11. Примерен изглед от модула за линеен анализ на резултатите от прототип на система за подпомагане вземането на решения.

Въпреки отчетените различия в разпределението на влиянието по групи, крайните агрегирани резултати при двата метода са сравнително близки.

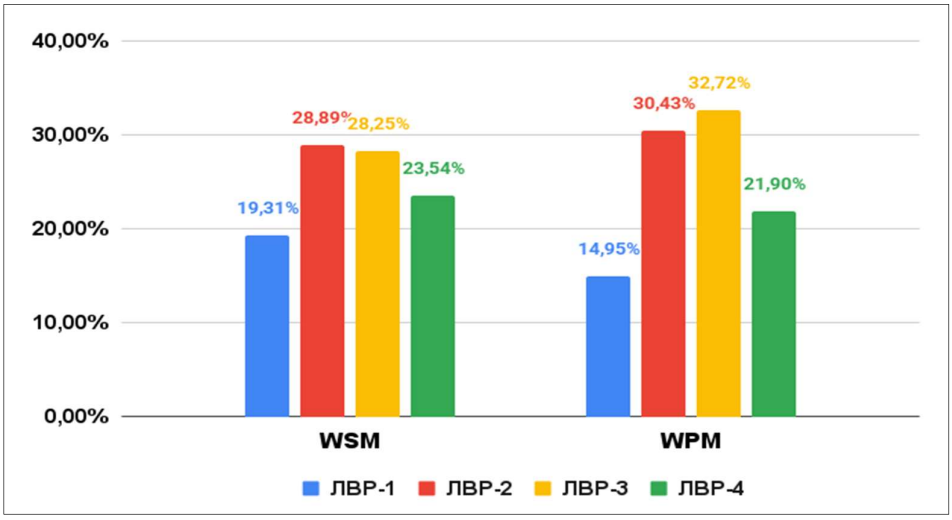
С цел да се отличи влиянието на ЛВР, е направено сравнение между агрегираните оценки общо за всички алтернативи при двата метода, илюстрирано на Фиг. 4.12.



Фиг. 4.12. Сравнение на влиянието на ЛВР върху агрегираните оценки за всички алтернативи, по групи критерии, при двата метода.

Разликите при Група 1 се дължат на характера на метода на претегленото произведение, при който се придава по-голямо влияние на лицата, които дават крайни за оценъчния интервал стойности (независимо дали са ниски или високи). Именно такива стойности на оценките са задали ЛВР-2 и ЛВР-3 по Група критерии 1 и алтернатива А1 (вж. Таблица 4.3). Изводът е, че при еднаква компетентност на ЛВР резултатите от WPM проявяват неговите характеристики.

Сравнение на влиянието на отделните ЛВР, общо за задачата, при двата метода е представено с диаграмата на Фиг.4.13.



Фиг. 4.13. Сравнение на влиянието на ЛВР при формирането на окончателното групово решение, между метода на претеглената сума и метода на претегленото произведение.

От тази фигура се забелязва, че и при двата метода, конкуренцията за доминация е между ЛВР-2 и ЛВР-3. Анализът сочи, че при WPM, ЛВР-3 получава по-висока степен на доминация в сравнение с ЛВР-2 в рамките на цялата задача. Този резултат, отново, може да се обясни с характерната чувствителност на мултипликативния модел към крайни стойности. В конкретния случай това се дължи на по-високото общо ниво на компетентност, присъдено на ЛВР-3 спрямо ЛВР-2 (вж. Фиг. 4.4.), в комбинация с това, че ЛВР-3 е лицето раздало по-високи оценъчни точки на алтернативите спрямо ЛВР-2 (вж. Фиг. 4.3.б).

4.2. Числено тестване на модификацията, формализираща генериране на оценки за алтернативите

За да се определи приложимостта на предложената модификация на метода на претеглената сума, формулирана чрез модела (2.13) – (2.21) (вж. Глава 2, т. 2.2), както и приложимостта на варианта на комбиниран модел (3.4), който включва същата модификация, е разгледана задача за избор на монитори, като част от офис оборудването на средна по размер компания от сферата на ИТ. Задачата е подходяща за експериментален пример, защото може да се идентифицират достатъчно критерии, за които са налични обективни данни, което е задължително изискване за прилагане на модификацията.

Изборът на подходящ монитор е предпоставка за комфортна и ефективна работа в офис среда. В съвременните фирми дигиталната трансформация устойчиво заменя конвенционалните ресурси. Това означава, че хардуерната база пряко влияе върху здравето на работещите. Различните автори установяват, че работата на устройства с екран [Fostervold и др., 2006], както и работата с множество монитори [Gallagher и др., 2019], често води до болки във врата или гърба поради лоша ергономия на работното място. Изследванията в тази област идентифицират синдром на компютърното зрение, който се изразява в дискомфорт в очите и главоболие, произтичащи от продължително гледане в екрани [Zheng и др., 2008]. Техниките за намаляване на този синдром включват всичко – от използването на подходящ хардуер до добър и ергономичен дизайн на работното пространство. Следователно, изборът на офис монитори е

актуална задача, спецификата на която е предпоставка тя да бъда решавана чрез предложения модифициран модел.

4.2.1. Сценарии за тестване

Разглеждат се три сценария в условията на групово вземане на решение. Всички те се развиват върху едни и същи коефициенти за важност на критериите и едни и същи обективни данни за алтернативите. Първите два сценария (Сценарий 1 и Сценарий 2) се базират на описаната в Глава 2, т. 2.2 модификация на метода WSM (2.13) – (2.21). При Сценарий 1, генерирането на оценка на база определяне на ползата от критериите (2.17), зависи изцяло от предпочитанията на администратора на задачата. При Сценарий 2 се демонстрира възможността за индивидуално определяне на ползата от критериите. В допълнение, при Сценарий 2 се тества и генерирането на оценки спрямо декомпозираните критерии при промяната в предпочитанията на ЛВР към опциите (2.18). При Сценарий 3, задачата се решава чрез комбиниране на модификацията за генериране на оценки с модификацията за отчитане на компетентността на ЛВР в различни области на компетентност, съгласно комбинирания модел (3.4). Входните данни при Сценарий 3 са аналогични на тези при Сценарий 2, с тази разликата, че се използва различна техника за нормализация. Целта при този сценарий е да се демонстрира и приложимостта на обобщения алгоритъм представен в Глава 3, т. 3.1 и гъвкавостта на проектираната система за подпомагане вземането на решения.

4.2.2. Формулиране на задачата за числено тестване

Задачата за определяне на подходящо офис оборудване се реализира при следните изхо́дни условия: 3 алтернативи, 3 ЛВР и 3 групи критерии, съдържащи общо 12 критерия, 4 от които са декомпозирани до 13 опции. Предварително определени като подходящи са следните 3 типа монитори (3 алтернативи): Dell U2724DE (**A1**), Samsung U32J590 (**A2**), MSI MEG 342C (**A3**). Лицата, формиращи крайното групово решение са както следва: (1) специалист от финансовия отдел (**ЛВР-1**); (2) специалист човешки ресурси (**ЛВР-2**); (3) техническо лице - софтуерен инженер (**ЛВР-3**). Критериите за избор на подходящия тип монитор са организирани в три тематично свързани групи, както е показано в Таблица 4.4.

Таблица 4.4. Групирани критерии за оценка на алтернативите, тип на критериите и коефициенти за тяхната важност определени от ЛВР.

Група критерии	Критерии	Тип на критерия	Коефициенти за важност на критериите		
			ЛВР-1	ЛВР-2	ЛВР-3
Група 1: Финансови критерии	C1: Цена	Числено представен	0,19	0,07	0,03
	C2: Гаранция	Числено представен	0,16	0,12	0,01
	C3: Консумация на енергия	Числено представен	0,11	0,07	0,01
Група 2: Грижа за здравето и ергономия	C4: Защита на зрението	Матрично представен	0,10	0,19	0,15
	C5: Ъгъл на наклон	Числено представен	0,04	0,07	0,05
	C6: Портретен режим	Числено представен	0,02	0,09	0,15
Група 3: Технически критерии	C7: Размер на екрана	Матрично представен	0,09	0,08	0,20
	C8: Тип на панела	Матрично представен	0,08	0,06	0,10
	C9: Резолюция	Матрично представен	0,09	0,06	0,10
	C10: Честота на опресняване	Числено представен	0,04	0,06	0,05
	C11: Яркост	Числено представен	0,04	0,06	0,05
	C12: Цветова гама	Числено представен	0,04	0,07	0,10

Отчитайки условието, задачата да се тества в комбинация с модификацията за отчитане на компетентността на ЛВР, съгласно обобщения алгоритъм, критериите се групират по област на компетентност, с цел подпомагане администрирането на задачата при Сценарий 3. Критерии, заедно с типа им и определените теглови коефициенти за важност на критериите, които са валидни еднакво и за трите сценария, са представени в Таблица 4.4.

Всички критерии от **Група 1** са от тип числено представени критерии. Типът на критериите е дефиниран при описанието на алгоритъма за реализация на модификацията в Глава 2, т. 2.2.1.

Много важни характеристики, които обединяват технологиите за опазване на зрението и опорно-двигателната система, чрез ергономията при употреба на монитори, са представени в **Група 2**. Критерий C4 е матрично представен и съдържа технологии за защита на зрението, декомпозирани като опции. Останалите критерии от тази група подпомагат правилната позиция на тялото и продуктивността на служителите.

Например, някои програмисти и дизайнери предпочитат да работят като използват портретен режим (C6).

От критериите с технически характер, съдържащи се в **Група 3**, половината се декомпозират до опции C7, C8, C9, а останалите се представят с числената си стойност.

Опциите, до които се декомпозират матрично представените критерии за конкретната задача са представени в Таблица 4.5.

Таблица 4.5. Матрично представени критерии декомпозирани до опции.

C4: Защита на зрението	
O1	Филтър за синя светлина
O2	Контрол над ШИМ
O3	Сензор за околна светлина
O4	Антирефлексно покритие
C7: Размер на екрана	
O5	Размер на екрана 27”
O6	Размер на екрана 31.5”
O7	Размер на екрана 34.2”
C8: Тип на панела	
O8	IPS
O9	VA
O10	OLED
C9: Резолюция	
O11	UW-QHD
O12	UHD
O13	QHD

С цел защитата на зрението, C4 е декомпозиран до четири основни технологии за минимизиране на дискомфорта от синята светлина, широко импулсната модулация, отраженията и неадаптивната яркост на екрана. Специфично за C7 е, че може да се изрази директно чрез размера в инчове, но е избрано да се декомпозира, за да се даде на ЛВР възможност да изразят предпочитанието си към конкретен размер екран.

При наличието на описаните до тук специфични особености на разглежданата задача за избор на подходящ тип монитор за средна ИТ компания, следва да се пристъпи към генерирането на оценки, за които както гласи модификацията са базирани на обективни данни и на изразените предпочитания на ЛВР.

За генерирането на оценките на алтернативите спрямо числено представените критерии $(b_{i,j}^{k})^{norm}$ (2.17), се използват данни от официалните спецификации, които са публично предоставени от производителите на монитори и актуалните търговски условия към момента на изследването. Данните, заедно с изразените от ЛВР предпочитания относно полезността по критериите, т.е. дали критерият се подобрява ако се максимизира или минимизира, са представени в Таблица 4.6.

Таблица 4.6. Числено представени критерии и техните мерни единици, стойности за алтернативите и предпочитания на ЛВР за полезността им при трите сценария.

Критерии	Мерна единица	Стойности за алтернативите			Сценарий 1 - предпочитания			Сценарий 2- предпочитания			Сценарий 3- предпочитания		
		A1	A2	A3	ЛВР1	ЛВР2	ЛВР3	ЛВР1	ЛВР2	ЛВР3	ЛВР1	ЛВР2	ЛВР3
C1	EUR	505	290	1247	cost	cost	cost	cost	cost	benefit	cost	cost	benefit
C2	Месеци	36	24	36	benefit	benefit	benefit	benefit	benefit	benefit	benefit	benefit	benefit
C3	Кв./Ч	24,4	59	42,3	cost	cost	cost	cost	cost	cost	cost	cost	cost
C5	Градуси	26	17	25	benefit	benefit	benefit	cost	benefit	benefit	cost	benefit	benefit
C6	Наличност	1	0	0	cost	cost	cost	cost	cost	benefit	cost	cost	benefit
C10	Hz	120	60	175	benefit	benefit	benefit	cost	benefit	benefit	cost	benefit	benefit
C11	cd/m²	350	270	250	cost	cost	cost	cost	cost	benefit	cost	cost	benefit
C12	% sRGB	100	99,1	139	benefit	benefit	benefit	cost	benefit	benefit	cost	benefit	benefit

При Сценарий 1, в Таблица 4.6. се забелязва, че предпочитанията на всяко ЛВР съвпадат, това се дължи на условието полезността на критериите да се определи от администратора на задачата. Изглежда, че стойностите по разглежданите критерии съвпадат за Сценарий 2 и Сценарий 3, но всъщност ще се приложи различна нормализационна техника и генерираните оценки ще се различават. Избрано е за Сценарий 1 и Сценарий 2 да се приложи линейна техника за нормализация на данните, а за Сценарий 3 – векторна техника за нормализация. Целта е да се демонстрира гъвкавостта на проектираната система, където се предлага избор на нормализационна техника.

За генерирането на оценките на алтернативите спрямо матрично представените критерии $c_{i,j}^{k}$ (2.18) – (2.21), също се използват данни от спецификациите на

производителите на монитори. Данните относно опциите, заедно с определените от ЛВР оценъчни точки за важност на опциите, са представени в Таблица 4.7.

Таблица 4.7. Матрично представени критерии, декомпозирани до опции и техните стойности за алтернативите заедно с определените от ЛВР оценъчни точки за важността им при трите сценария.

Критерии		Стойности на опциите			Оценъчни точки за важността на опциите								
					Сценарий 1			Сценарий 2			Сценарий 3		
№	Опции	A1	A2	A3	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-1	ЛВР-2	ЛВР-3
C4	O1	1	0	1	4	5	5	4	5	5	4	5	5
	O2	0	1	1	4	5	5	4	5	5	4	5	5
	O3	1	0	1	2	2	4	2	2	4	2	2	4
	O4	1	1	1	2	3	5	2	3	5	2	3	5
C7	O5	1	0	0	4	3	3	4	3	**1**	4	3	1
	O6	0	1	0	4	3	4	4	3	**5**	4	3	5
	O7	0	0	1	2	3	5	2	3	**3**	2	3	3
C8	O8	1	0	0	4	4	2	4	4	2	4	4	2
	O9	0	1	0	4	3	3	4	3	**1**	4	3	1
	O10	0	0	1	2	2	5	2	2	5	2	2	5
C9	O11	0	0	1	1	2	5	1	2	5	1	2	5
	O12	0	1	0	2	5	5	2	5	**2**	2	5	2
	O13	1	0	0	4	3	2	4	3	2	4	3	2

Трябва да се отбележи, че опциите приемат булева стойност, която отразява наличието на съответната характеристика при всяка една от алтернативите. ЛВР изразяват предпочитанията относно опциите чрез скалата от 1 до 5, като 1 изразява най-ниска степен на важност, а 5 изразява важност от най-висока степен.

В съответствие със Сценарий 3, за определените групи критерии са зададени теглови коефициенти, отразяващи степента на компетентност на ЛВР, както е показано в Таблица 4.8.

Таблица 4.8. Теглови коефициенти за компетентност на ЛВР, спрямо групите от критерии.

Групи критерии	ЛВР-1	ЛВР-2	ЛВР-3
Група 1: Финансови критерии	0.62	0.21	0.17
Група 2: Грижа за здравето и ергономия	0.26	0.40	0.34
Група 3: Технически критерии	0.30	0.15	0.55

За всеки от трите описани сценария, съобразно входните данни се генерират индивидуални формуляри за попълване на предпочитанията на ЛВР, въз основа на които се изчисляват оценките за алтернативите. Данните се обединяват в матрици за групово вземане на решение, върху които се прилага модела на претеглената сума, за формиране на колективно решение за избор на монитор.

4.2.3. Резултати от численото тестване на модификацията на метода на претеглената сума

Модификацията засяга алгоритъма за реализация на метода на претеглената сума, като формализира подход за генериране на оценки, без да променя механизма за агрегиране на атрибутите в крайния резултат. Поради това интерпретацията на резултатите е насочена към стъпките за генериране на оценки в средата на разработения прототип на система за подпомагане вземането на решения.

Съгласно **Стъпка 4** от алгоритъма за реализиране на модификацията (вж. Глава 2, т. 2.2.1), се генерира формуляр за въвеждане на предпочитания към числените критерии. При Сценарий 1 формулярът се попълва единствено от администратора, като въведените предпочитания се отнасят за всички ЛВР. Изглед от формуляра е показан на Фиг. 4.14.

Фиг. 4.14. Формуляр за изразяване на предпочитания към полезността по числените критерии. Изглед от прототипът на система за подпомагане вземането на решения.

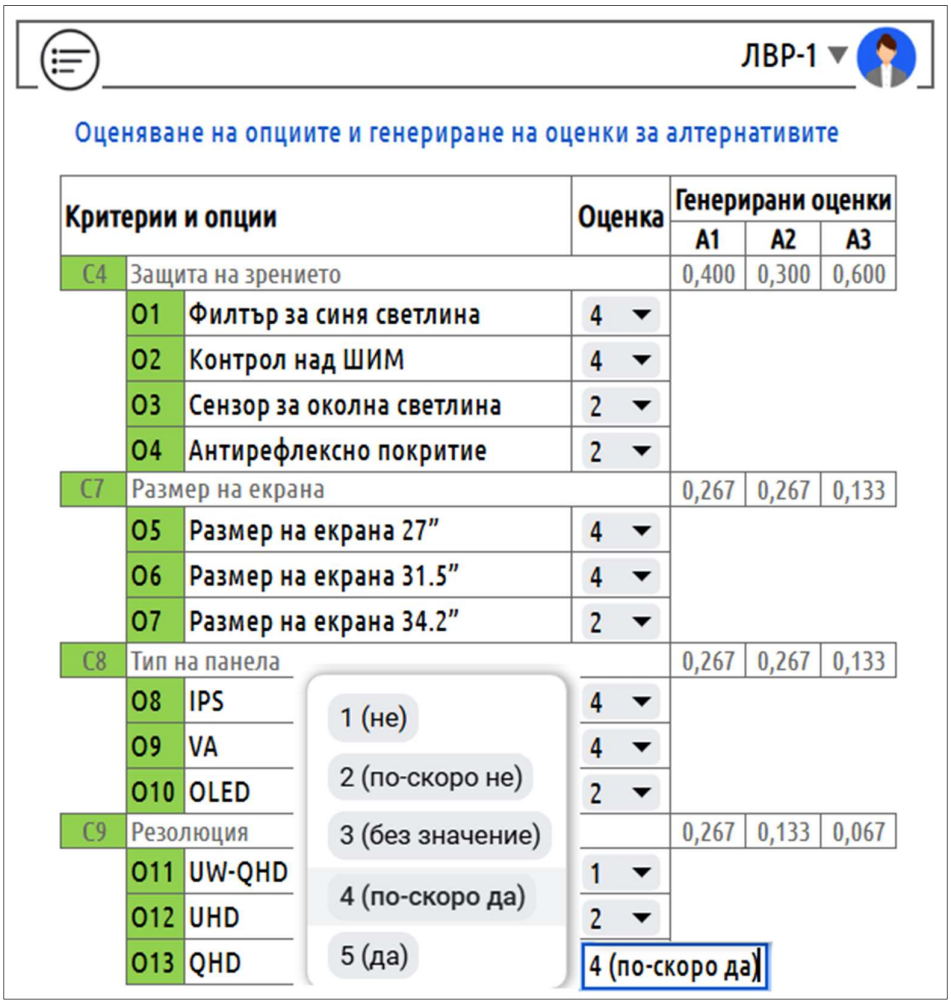
При така изразените предпочитания, на база входните данни (вж. Таблица 4.6) за всяка от алтернативите се изчислява оценката $(b_{i,j}^{k})^{norm}$ (2.17), в съответствие с избраната нормализационна техника. В случая, за Сценарий 1 е избрано да се приложи линейна нормализация, използваща Max и Min (1.12.а) и (1.12.б) върху данните на числено представените критерии, което се демонстрира със следните примери за генериране на оценки на алтернатива A1 по критерии C1 и C2:

$$(b_{1,1}^{1,2,3})^{cost} = \frac{\min\ (505, 290, 1247)}{505} = 0{,}574$$

$$(b_{1,2}^{1,2,3})^{benefit} = \frac{36}{\max(36, 24, 36)} = 1{,}000$$

На същия принцип се изчисляват оценките и за останалите алтернативи.

Когато критерият е от тип матрично представен, съгласно **Стъпка 7** от алгоритъма, на всеки ЛВР се предоставя възможност да изрази предпочитанията си към опциите на алтернативите. За тази цел в прототипа на системата за подпомагане вземането на решения се генерира индивидуален формуляр за всяко ЛВР. Примерен изглед, при Сценарий 1, за ЛВР-1 е представен на Фиг. 4.15.



Фиг. 4.15. Формуляр за изразяване на предпочитания към опциите на алтернативите, при матрично представените критерии. Изглед от прототипът на система за подпомагане вземането на решения.

Предпочитанията се изразяват по скала, отговаряйки на въпроса: „До каква степен искате опцията да е налична в алтернативата?“. Въз основа на данните от Таблица 4.7, по трите атрибута (ЛВР, алтернатива, критерий) се изчислява оценката $c_{i,j}^{k}$ (2.18), при спазване на ограниченията (2.19) и (2.21). Примерно генериране на оценка от тип $c_{i,j}^{k}$, се дава с изчислението на оценката на алтернатива A1, спрямо критерий C4, направена от ЛВР-1, както следва.

$$c_{1,4}^{1} = \frac{4*1+4*0+2*1+2*1}{4*5} = 0{,}400$$

Следвайки примерът, се изчисляват и останалите оценки.

Така генерираните оценки, заедно с коефициентите за важност на критериите са обобщени в матрицата за вземане на решение от Фиг. 4.16.

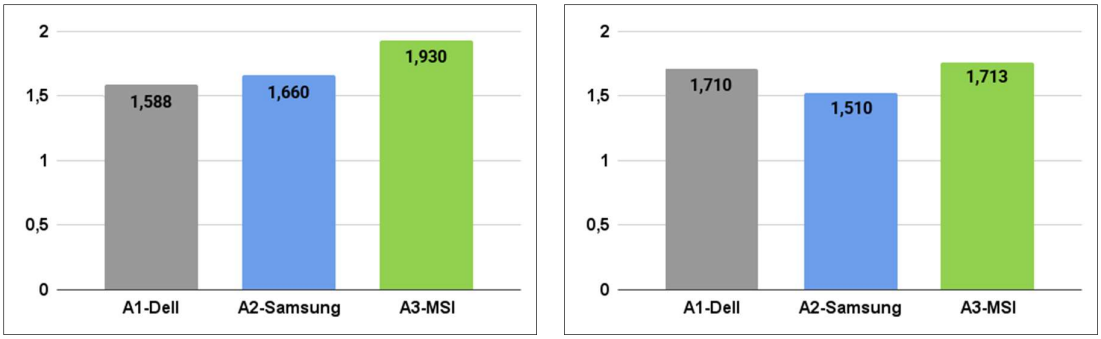
Администратор

Обобщена матрица на алтернативите

КРИТЕРИИ		Теглови коефициенти за важност на критериите			Генерирани оценки за алтернативите								
					A1 Dell U2724DE			A2 Samsung U32J590			A3 MSI MEG 342C		
		ЛВР-1: Финансист	ЛВР-2: Човешки ресурси	ЛВР-3: Софтуерен инженер	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-1	ЛВР-2	ЛВР-3
C1	Цена	0,19	0,07	0,03	0,574			1,000			0,233		
C2	Гаранция	0,16	0,12	0,01	1,000			0,667			1,000		
C3	Консумация на енергия	0,11	0,07	0,01	1,000			0,414			0,577		
C4	Защита на зрението	0,10	0,19	0,15	0,400	0,500	0,700	0,300	0,400	0,500	0,600	0,750	0,950
C5	Ъгъл на наклон	0,04	0,07	0,05	1,000			0,654			0,962		
C6	Портретен режим	0,02	0,09	0,15	0,000			1,000			1,000		
C7	Размер на екрана	0,09	0,08	0,20	0,267	0,200	0,200	0,267	0,200	0,267	0,133	0,200	0,333
C8	Тип на панела	0,08	0,06	0,10	0,267	0,267	0,133	0,267	0,200	0,200	0,133	0,133	0,333
C9	Резолюция	0,09	0,06	0,10	0,267	0,200	0,133	0,133	0,333	0,333	0,067	0,133	0,333
C10	Честота на опресняване	0,04	0,06	0,05	0,686			0,343			1,000		
C11	Яркост	0,04	0,06	0,05	0,714			0,926			1,000		
C12	Цветова гама	0,04	0,07	0,10	0,719			0,713			1,000		

Фиг. 4.16. Матрица за вземане на решение при Сценарий 1. Изглед от прототипът на система за подпомагане вземането на решения.

След прилагане на моделът на модифицирания метод на претеглената сума (2.13) върху матрицата за вземане на решение, за Сценарий 1, се агрегират резултати, класиращи алтернативите, така както е показано на Фиг. 4.17.а.



а) Сценарий 1 б) Сценарий 2

Фиг. 4.17. Класиране на мониторите при Сценарий 1 и Сценарий 2.

Оказва се, че на предпочитанията на ЛВР и на администратора, най-много отговаря монитор MSI (A3) с осезаема преднина, следван от Samsung (A2) и Dell (A1). За разлика, при Сценарий 2 класацията се променя като A3 запазва преднина, но тя е

значително скъсена от A1-Dell, който изпреварва A2-Samsung, както е илюстрирано на Фиг. 4.17.б.

Оценките за алтернативите, използвани за агрегиране на резултатите при Сценарий 2 са генерирани в съответствие с индивидуално изразените предпочитания на ЛВР и обективните данни от Таблица 4.6 и Таблица 4.7. Матрицата за вземане на решение е на Фиг. 4.18.

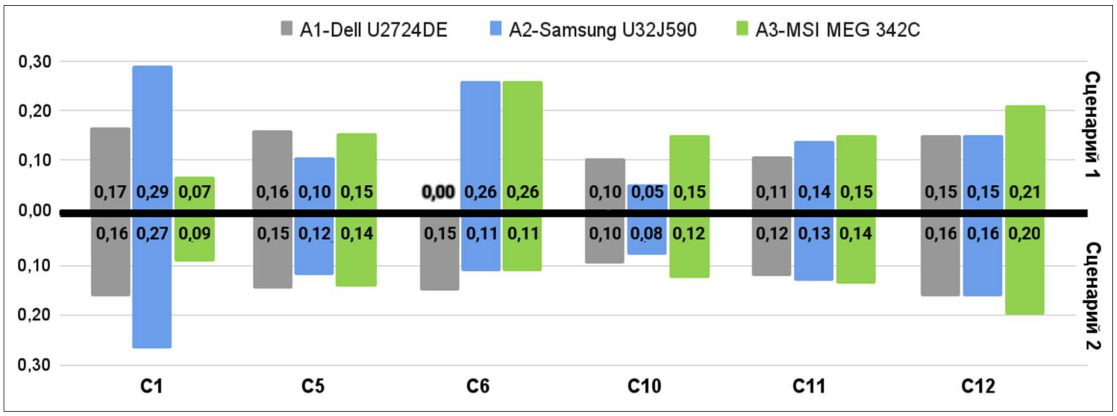
Администратор

Обобщена матрица на алтернативите

КРИТЕРИИ		Теглови коефициенти за важност на критериите			Генерирани оценки за алтернативите								
					A1 Dell U2724DE			A2 Samsung U32J590			A3 MSI MEG 342C		
		ЛВР-1: Финансист	ЛВР-2: Човешки ресурси	ЛВР-3: Софтуерен инженер	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-1	ЛВР-2	ЛВР-3
C1	Цена	0,19	0,07	0,03	0,574	0,574	**0,405**	1,000	1,000	**0,233**	0,233	0,233	**1,000**
C2	Гаранция	0,16	0,12	0,01	1,000	1,000	1,000	0,667	0,667	0,667	1,000	1,000	1,000
C3	Консумация на енергия	0,11	0,07	0,01	1,000	1,000	1,000	0,414	0,414	0,414	0,577	0,577	0,577
C4	Защита на зрението	0,10	0,19	0,15	0,400	0,500	0,700	0,300	0,400	0,500	0,600	0,750	0,950
C5	Ъгъл на наклон	0,04	0,07	0,05	**0,654**	1,000	1,000	**1,000**	0,654	0,654	**0,680**	0,962	0,962
C6	Портретен режим	0,02	0,09	0,15	0,000	0,000	**1,000**	1,000	1,000	**0,000**	1,000	1,000	**0,000**
C7	Размер на екрана	0,09	0,08	0,20	0,267	0,200	**0,067**	0,267	0,200	**0,333**	0,133	0,200	**0,200**
C8	Тип на панела	0,08	0,06	0,10	0,267	0,267	0,133	0,267	0,200	**0,067**	0,133	0,133	0,333
C9	Резолюция	0,09	0,06	0,10	0,267	0,200	0,133	0,133	0,333	**0,133**	0,067	0,133	0,333
C10	Честота на опресняване	0,04	0,06	0,05	**0,500**	0,686	0,686	**1,000**	0,343	0,343	**0,343**	1,000	1,000
C11	Яркост	0,04	0,06	0,05	0,714	0,714	**1,000**	0,926	0,926	**0,771**	1,000	1,000	**0,714**
C12	Цветова гама	0,04	0,07	0,10	**0,991**	0,719	0,719	**1,000**	0,713	0,713	**0,713**	1,000	1,000

Фиг. 4.18. Матрица за вземане на решение при Сценарий 2. Изглед от прототипът на система за подпомагане вземането на решения.

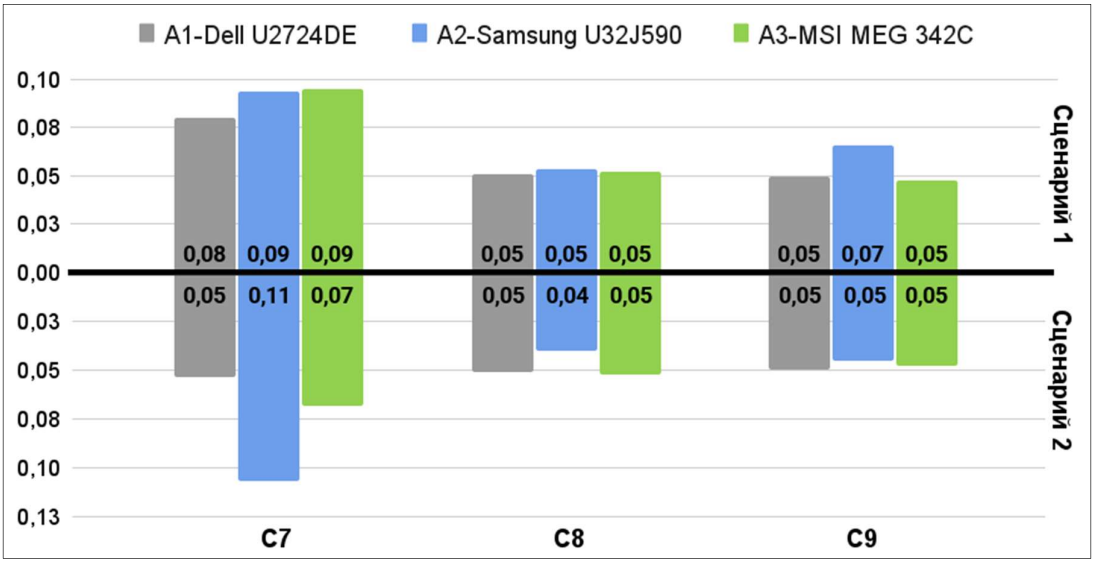
Анализирайки Фиг. 4.17 се установява, че като цяло агрегираните резултати спадат при A2 (със 9,0%) и A3 (с 11,2%) за сметка на A1-Dell, който се покачва с почти 8 % (спрямо Сценарий 1). Очаквано, промяната в резултатите се дължи на натрупващия се ефект от променените индивидуални предпочитания на ЛВР по девет от критериите. Интересно е да се направи сравнение на представянето на алтернативите по числено представените критерии C1, C5, C6, C10, C11 и C12 при двата сценария, което е показано на Фиг. 4.19.



Фиг. 4.19. Сравнение на представянето на алтернативите по числено представените критерии, за които има променено предпочитание при Сценарий 2.

От Фиг.4.19 се забелязва, че алтернатива А3 губи точки по повечето критерии. Най-голяма промяна се забелязва при критерий С6, където може да се потърси конкретната причина за получения резултат. При дадената от ЛВР-3 висока важност на критерий С6 и факта, че само монитор Dell поддържа портретен режим обръщат резултата.

Аналогично сравнение за матрично представените критерии С7, С8 и С9, е представено на Фиг. 4.20.



Фиг. 4.20. Сравнение на представянето на алтернативите по матрично представените критерии, за които има променено предпочитание при Сценарий 2.

Категоричното мнение против VA панела и UHD резолюцията от страна на софтуерния инженер подсилва занижения резултат на монитор Samsung, тъй като единствено той ги поддържа, но пък предпочетения размер на екрана е в негова полза.

4.2.4. Резултати от численото тестване на комбинирания модел на метода на претеглената сума

При Сценарий 3, променяйки нормализационната техника и включвайки тегловите коефициенти за компетентност на ЛВР, матрицата за групово вземане на решение, придобива различен вид, както е илюстрирано на Фиг. 4.21.

Администратор

Обобщена матрица на алтернативите

КРИТЕРИИ		Теглови коефициенти за важност на критериите и за компетентност на ЛВР			Генерирани оценки за алтернативите								
					A1 Dell U2724DE			A2 Samsung U32J590			A3 MSI MEG 342C		
		ЛВР-1: Финансист	ЛВР-2: Човешки ресурси	ЛВР-3: Софтуерен инженер	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-1	ЛВР-2	ЛВР-3	ЛВР-1	ЛВР-2	ЛВР-3
Група 1: Финансови критерии		0,62	0,21	0,17	0,224			0,179			0,136		
C1	Цена	0,19	0,07	0,03	**0,633**	**0,633**	**0,367**	**0,789**	**0,789**	**0,211**	**0,094**	**0,094**	**0,906**
C2	Гаранция	0,16	0,12	0,01	**0,640**	**0,640**	**0,640**	**0,426**	**0,426**	**0,426**	**0,640**	**0,640**	**0,640**
C3	Консумация на енергия	0,11	0,07	0,01	**0,681**	**0,681**	**0,681**	**0,230**	**0,230**	**0,230**	**0,448**	**0,448**	**0,448**
Група 2: Грижа за здравето и...		0,26	0,40	0,34	0,168			0,130			0,194		
C4	Защита на зрението	0,10	0,19	0,15	0,400	0,500	0,700	0,300	0,400	0,500	0,600	0,750	0,950
C5	Ъгъл на наклон	0,04	0,07	0,05	**0,348**	**0,652**	**0,652**	**0,574**	**0,426**	**0,426**	**0,373**	**0,627**	**0,627**
C6	Портретен режим	0,02	0,09	0,15	**0,000**	**0,000**	**1,000**	**1,000**	**1,000**	**0,000**	**1,000**	**1,000**	**0,000**
Група 3: Технически критерии		0,30	0,15	0,55	0,139			0,154			0,177		
C7	Размер на екрана	0,09	0,08	0,20	0,267	0,200	0,067	0,267	0,200	0,333	0,133	0,200	0,200
C8	Тип на панела	0,08	0,06	0,10	0,267	0,267	0,133	0,267	0,200	0,067	0,133	0,133	0,333
C9	Резолюция	0,09	0,06	0,10	0,267	0,200	0,133	0,133	0,333	0,133	0,067	0,133	0,333
C10	Честота на опресняване	0,04	0,06	0,05	**0,456**	**0,544**	**0,544**	**0,728**	**0,272**	**0,272**	**0,206**	**0,794**	**0,794**
C11	Яркост	0,04	0,06	0,05	**0,311**	**0,311**	**0,689**	**0,468**	**0,468**	**0,532**	**0,508**	**0,508**	**0,492**
C12	Цветова гама	0,04	0,07	0,10	**0,495**	**0,505**	**0,505**	**0,499**	**0,501**	**0,501**	**0,297**	**0,703**	**0,703**

Фиг. 4.21. Матрица за вземане на решение при Сценарий 3, с включени агрегирани оценки по групи критерии. Изглед от прототипът на система за подпомагане вземането на решения.

За Сценарий 3 е избрано да се приложи векторна нормализация (1.14.а) и (1.14.б) върху данните на числено представените критерии (вж. Таблица 4.6). Спазвайки

условията на сценария, се дават следните примери за изчисляване на оценки на алтернатива А1 по критерии С1 (за ЛВР-1 и ЛВР-2) и по С2 (за всички ЛВР):

$$(b_{1,1}^{1,2})^{cost} = 1 - \frac{505}{\sqrt{505^2 + 290^2 + 1247^2}} = 0{,}633$$

$$(b_{1,2}^{1,2,3})^{benefit} = \frac{36}{\sqrt{36^2 + 24^2 + 36^2}} = 0{,}640$$

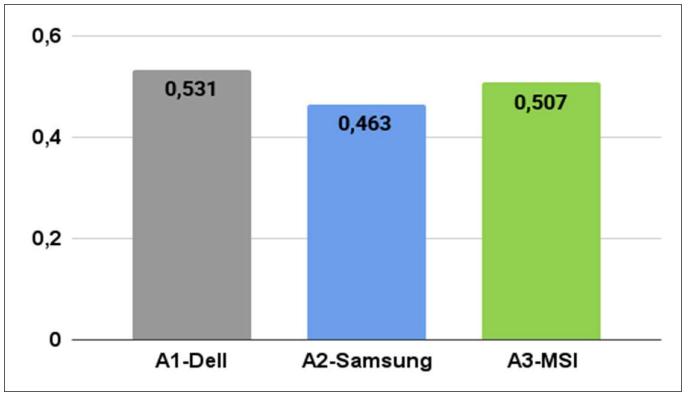
По същия начин се изчисляват оценките и за останалите алтернативи.

За агрегирането на оценките съгласно комбинирания модел от (3.4), се дава за пример изчислението на оценката на алтернатива А1 по Група критерии 1 (виж резултата на Фиг. 4.21), както следва:

$$A_{1,\text{Група 1}} = 0{,}62 * (0{,}19 * 0{,}633 + 0{,}16 * 0{,}640 + 0{,}11 * 0{,}681) +$$
$$+ 0{,}21 * (0{,}07 * 0{,}633 + 0{,}12 * 0{,}640 + 0{,}7 * 0{,}681) +$$
$$+ 0{,}17 * (0{,}03 * 0{,}367 + 0{,}01 * 0{,}640 + 0{,}01 * 0{,}681) = 0{,}224$$

За да се получи финалната оценка на тази и всички останали алтернативи, резултатите по групи критерии се сумират.

Установените тенденции при изхо́дния Сценарий 2 (агрегираната оценка на А1 да расте, за сметка на тези на А2 и А3) се потвърждават и дори увеличават при Сценарий 3, което води до промяна в класацията, като А1 успява да надмине А3, както е показано на Фиг. 4.22.



Фиг. 4.22. Класиране на мониторите при Сценарий 3.

Тествани поотделно и двете промени – добавяне на коефициенти за компетентност и векторна нормализация, водят до една и съща класация на алтернативите. Което показва, че гъвкавостта на системата за подпомагане вземането на решения позволява напасване към различни задачи.

Двете модификации се съчетават и допълват успешно, тъй като не си пречат, изпълнявайки се на различни стъпки от алгоритъма.

4.3. Числено тестване на модификацията, формализираща генериране на коефициенти, даващи предимство в общото представяне на алтернативите

Приложимостта на предложената модификация на WSM и WPM, формализираща генерирането на теглови коефициенти за алтернативите (вж. Глава 2, т. 2.3), е тествана при задача за избор на софтуерна рамка за уеб разработка. Задачата е подходяща за валидиране на приложимостта на модификацията, тъй като може лесно и интуитивно да се идентифицира ключов критерий (напр. език за програмиране) спрямо, който да се претеглят рамките за уеб разработка. В допълнение могат да се включат различни типове критерии – както обективни, например рейтинг, производителност, честота на актуализации, така и субективни, например лекота на използване, гъвкавост или предпочитание сред екипа. Това позволява да се тества и обобщеният модел, съчетаващ всички описани в настоящия труд модификации (вж. Глава 3, т. 3.1).

Рамките за уеб разработка (web development frameworks) представляват съществен елемент от съвременното софтуерно инженерство. Те осигуряват структуриран подход за създаване на динамични и мащабируеми уеб приложения. Често интегрират разнообразни библиотеки и инструменти, включително такива за автоматично генериране на код (напр. потребителски интерфейси, компоненти и шаблони), за осигуряване на информационна сигурност (напр. автентикация, управление на сесии, криптиране на комуникацията), както и за достъп и работа с бази данни, и API за външни услуги. Съобразно архитектурната структура на уеб приложенията, рамките за разработка обикновено са специализирани в един от двата слоя – frontend (слой на потребителския интерфейс) или backend (сървърна логика и обработка на данни), като съществуват и full-stack рамки, обединяващи двете функционалности. Основните им предназначения са: (1) да облекчат екипа от разработчици, като предоставят готови модули и инструменти; (2) да насърчават добри практики в програмирането; (3) да ускоряват процеса по разработка [Laaziri и др., 2019]. Следователно чрез тях се търси повишаване производителността в разработването на

софтуер и подобрение в качеството на програмния код [Sadowski и др. , 2019; Wagner и Deissenboeck, 2019].

Темата е актуална, предвид динамичната еволюция на уеб технологиите и нарастващото разнообразие от налични рамки за уеб разработка, които се конкурират по близки характеристики.

4.3.1. Сценарии за тестване

За да се тества приложимостта на предложената модификация върху моделите и алгоритмите на: 1) метода на претеглената сума; 2) метода на претегленото произведение и 3) обобщения модел на метода на претеглената сума, съчетаващ предложените в настоящия дисертационен труд модификации, са съставени три сценария. Сценарий 1 и Сценарий 2 се развиват в условието на индивидуално вземане на решение и имат за цел да изследват ефекта от нововъведения тип коефициент при използване на методите WSM и WPM. За сравнение са използвани резултатите от класическите методи. Разликата между Сценарий 1 и Сценарий 2, е в промяната на предпочитанията към опциите при ключовия критерий, използван за да даде предимство на алтернативите, а също така и при определените коефициенти за важност на останалите критерии.

Сценарият, използван за тестване на приложимостта на обобщения модел (Сценарий 3) се развива в условието на групово вземане на решение. При него параметрите на задачата се надграждат, като се добавят още два допълнителни критерия – единият, от които е матрично представен (декомпозиращ се до опции), а другият критерий е субективен. По този начин се обхваща пълната гама от възможности, които дават модификациите, имплементирани в прототипа на СПВР.

4.3.2. Формулиране на задачата за числено тестване

Задача за избор на рамка за разработка на уеб приложения се оразмерява при първите два сценария със следните параметри: 6 алтернативи, 1 лице, вземащо решение, общо 3 критерия, като 2 от тях се използват за оценяване, а другия е за предимство на алтернативите и се декомпозира до 3 опции. При Сценарий 3, алтернативите се запазват, а ЛВР стават двама. Критериите стават 5, защото се добавят – един субективен

и един матрично представен критерий, така че да се покрият всички типове оценки, допустими при обобщения алгоритъм.

Шест популярни рамки за уеб разработка са избрани за оценяване с цел определяне на най-подходящата алтернатива за конкретен екип от разработчици. Те включват: React (**A1**), Vue.js (**A2**), Django (**A3**), Flask (**A4**), Laravel (**A5**) и Symfony (**A6**).

Лицата, вземащи решение са софтуерни инженери от един екип, но с различно ниво на опит. Старши софтуерен инженер (**ЛВР-1**) участва при Сценарий 1 и Сценарий 3 и младши софтуерен инженер (**ЛВР-2**), който участва в Сценарий 2 и Сценарий 3.

Петте критерия, използвани в задачата, се разграничават по два признака съгласно обобщения алгоритъм (Глава 3, т. 3.1). Първо – според предназначението им: (1) за оценяване или (2) за предимство на алтернативи (вж. Глава 2, т. 2.3.1). Второ – според начина на формиране на оценките: (1) субективни, (2) числено представени или (3) матрично представени, с възможност за декомпозиране до опции (вж. Глава 2, т. 2.2.1). Към типовете са включени и тегловите коефициенти за важност на критериите в трите сценария. Всички тези входни данни са обобщени в Таблица 4.9.

Таблица 4.9. Типове критерии – за оценяване и за предимство на алтернативите. Коефициенти за важност на критериите, определени от ЛВР, за трите сценария.

Критерии		**Тип на критериите, спрямо предназначението и спрямо начина на формиране на оценката**	**Коефициенти за важност на критериите**			
			Сценарий 1	**Сценарий 2**	**Сценарий 3**	
			ЛВР-1	**ЛВР-2**	**ЛВР-1**	**ЛВР-2**
C1	Език за програмиране	За предимство на алтернативи/ Матрично представен	-	-	-	-
C2	GitHub класация	За оценяване/ Числено представен	0,60	0,27	0,26	0,10
C3	StackOverflow класация	За оценяване/ Числено представен	0,40	0,73	0,17	0,32
C4	Покритие на софтуерни слоеве	За оценяване/ Матрично представен	x	x	0,41	0,16
C5	Лекота за работа	За оценяване/ Субективен	x	x	0,16	0,42

Езикът за програмиране е критерий от критична важност, тъй като отразява готовността и уменията на екипа за ефективно използване на рамката. Затова критерият C1 се използва за генериране на коефициенти, даващи предимство в общото

представяне на алтернативите и съответно за него не се определя коефициент за важност, както се вижда от Таблица 4.9. GitHub класацията (C2) отразява броя звезди (stars), които даден проект получава от потребителите на GitHub – най-голямата платформа за управление на версии на програмен код и сътрудничество върху отворени проекти. По-висок рейтинг предполага повече потребители, по-голямо доверие от страна на разработчиците и по-надеждна поддържка. Stack Overflow класацията (C3) отчита броя въпроси и дискусии, засягащи дадена рамка, зададени в най-голямата интернет общност за програмисти. Той е показателен за разпространението на технологията в практиката. Високият рейтинг косвено води до значителна база от вече решени проблеми, което е от голяма важност за един младши софтуерен инженер. Критерият C4 показва, кои архитектурни слоеве са обхванати от съответната рамка. Критерият C5 има за цел да отрази субективното усещане за това, колко лесна би била разработката на софтуер с всяка от рамките, въз основа на личния опит на ЛВР.

Тъй като критериите C1 и C4 са дефинирани като матрично представени критерии, те се декомпозират до опции, както е показано в Таблица 4.10.

Таблица 4.10. Матрично представени критерии (декомпозируеми до опции), използвани за генериране на оценка или за генериране на коефициент за предимство на алтернативите.

Критерии и наличните опции			Стойности на опциите						Коефициенти / Оценъчни точки за важност на опциите			
									Сценарий 1	Сценарий 2	Сценарий 3	
			A1	A2	A3	A4	A5	A6	ЛВР-1	ЛВР-2	ЛВР-1	ЛВР-2
C1	Език за програмиране											
	O1	JavaScript	1	1	0	0	0	0	0,46	0,22	0,46	0,22
	O2	Python	0	0	1	1	0	0	0,33	0,32	0,33	0,32
	O3	PHP	0	0	0	0	1	1	0,21	0,46	0,21	0,46
C4	Покритие на софтуерни слоеве											
	O4	Потребителски интерфейс	1	1	1	1	1	1	x	x	4	5
	O5	Клиентска логика	1	1	0	0	0	0	x	x	5	2
	O6	Сървърна логика	0	0	1	1	1	1	x	x	2	3
	O7	Работа с база данни	0	0	1	0	1	1	x	x	2	1

Спазвайки ограничението (2.19) и (2.25), опциите приемат булева стойност, която отразява наличието на съответната характеристика при всяка една от алтернативите. Важно е да се отбележи, че за опциите на C1, ЛВР определят

относителна важност на опциите чрез теглови коефициенти, съобразно ограничение (2.24). Докато за опциите на C4, ЛВР определят важността на опциите чрез задаване на оценъчни точки. Предпочитанията се изразяват чрез предварително определена скала, в случая от 1 (за най-малко предпочитана опция) до 5 (за най-много предпочитаната опция). Тъй като C4 не участва при първите два сценария, там липсват оценъчни точки за неговите опции.

Въз основа на обективните данни от класациите на рамките в GitHub и StackOverflow могат да бъдат формирани оценки в Сценарий 1 и Сценарий 2. При Сценарий 3, чрез изразяване на предпочитания относно тяхната полза или вреда, се генерират съответните оценки. Данните и за двата критерия (C2 и C3), са извлечени от специализирания сайт https://hotframeworks.com, който прави сравнения между различни рамки за уеб разработка и са представени в Таблица 4.11.

Таблица 4.11. Числено представени критерии, стойности за алтернативите и предпочитания на ЛВР за полезността на критериите при Сценарий 3.

Критерии	Стойности за алтернативите						Cost / Benefit
	A1 React	A2 Vue.js	A3 Django	A4 Flask	A5 Laravel	A6 Symfony	
C2: GitHub класация	99	100	89	89	90	81	benefit
C3: StackOverflow класация	97	87	97	83	93	86	benefit

Тегловите коефициенти за компетентност, които са определени за двете лица, вземащи решение, при Сценарий 3, са показани в Таблица 4.12.

Таблица 4.12. Теглови коефициенти за компетентност на ЛВР, спрямо отделните критерии, използвани при Сценарий 3.

Критерии	Теглови коефициенти за компетентност на ЛВР	
	ЛВР-1	ЛВР-2
C1: Език за програмиране	-	-
C2: GitHub рейтинг	0,70	0,30
C3: StackOverflow рейтинг	0,45	0,55
C4: Покритие на софтуерни слоеве	0,80	0,20
C5: Лекота за работа	0,50	0,50

За всеки от трите разгледани сценария, в съответствие с входните данни, се генерират индивидуализирани формуляри за попълване на предпочитанията на ЛВР. При Сценарий 1 и Сценарий 2, агрегирането на резултатите се извършва чрез прилагане на модифицираните версии на WSM и WPM. В Сценарий 3, наличните данни се интегрират в матрица за групово вземане на решение, върху която се прилага обобщеният модел за формиране на колективно решение относно избора на рамка за уеб разработка.

4.3.3. Резултати от численото тестване на модификацията на метода на претеглената сума

Численото тестване на модификацията, формализираща генериране на теглови коефициенти, даващи предимство в общото представяне на алтернативите, при метода на претеглената сума (Сценарий 1), се изпълнява в прототипа на СПВР. Матрицата за индивидуално вземане на решение, върху която се провежда експеримента, е представена на Фиг. 4.23.

ЛВР-1 (Старши софтуерен инженер)

Индивидуална матрица на алтернативите

КРИТЕРИИ		Теглови коефициенти за важност на критериите	Оценки за алтернативите / Коефициенти за коригиране на резултатите. По алтернативи					
			A1 React	A2 Vue.js	A3 Django	A4 Flask	A5 Laravel	A6 Symfony
			Коефициенти за коригиране на резултатите на алтернативите					
C1	Език за програмиране	-	0,46	0,46	0,33	0,33	0,21	0,21
			Оценки за алтернативите					
C2	GitHub класация	0,60	0,99	1,00	0,89	0,89	0,90	0,81
C3	StackOverflow класация	0,40	0,97	0,87	0,97	0,83	0,93	0,86

Фиг. 4.23. Матрица за индивидуално вземане на решение при Сценарий 1. Изглед от прототипът на система за подпомагане вземането на решения.

Съгласно Стъпка 8 от алгоритъма за прилагане на модификацията (вж. Глава 2, т. 2.3.1), за всяка алтернатива спрямо ключовия критерий C1 се генерира съответният коефициент ψ_i^k. Изчисленията се извършват на база входните данни, представени в Таблица 4.10, и в съответствие с формула (2.23). Следният пример илюстрира изчислението на коефициента ψ_1^1 за алтернатива A1-React:

$$\psi_1^1 = \sum_{n=1}^{1} \sum_{r_n=1}^{3} p_{r_n}^1 v_{1,r_n} = 0{,}46 * 1 + 0{,}33 * 0 + 0{,}21 * 0 = 0{,}46$$

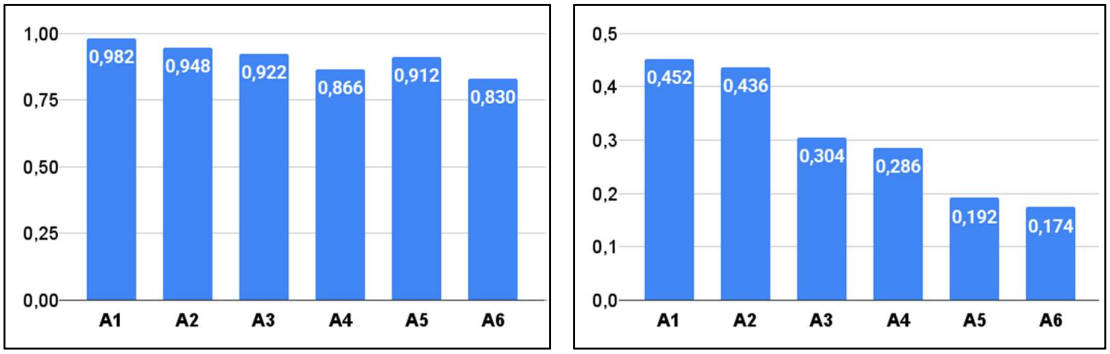
Идентично са получени коефициентите за останалите алтернативи.

Агрегирането на данните от матрицата за вземане на решение, за всяка алтернатива се извършва чрез прилагане на модифицирания модел на метода на претеглената сума (2.22). Пример за изчисление на крайния резултат за алтернатива A1-React е представен както следва:

$$A_{1\,WSM}^{GDM} = \sum_{k=1}^{1} \left(\psi_1^k \sum_{j=1}^{2} w_j^k \, a_{1,j}^k \right) = 0{,}46 * (0{,}60 * 0{,}99 + 0{,}40 * 0{,}97) \approx 0{,}452$$

Агрегираните резултати на другите алтернативи са изчислени по същия начин.

За да се демонстрира ефектът от нововъведените коефициенти, са изчислени резултати на алтернативите спрямо критерии C2 и C3, използвайки класическия метод на претеглената сума (1.3). Крайното класиране на алтернативите при двата метода е представено на Фиг. 4.24 (а, б).



а) Класически WSM б) Модифициран WSM

Фиг. 4.24. Класиране на рамките за уеб разработка при Сценарий 1.

Най-предпочитана е алтернатива A1 (React), което е очаквано тъй като тя получава най-висока сумарна оценка по двата използвани критерия за оценяване. се откроява ясно във Фиг. 4.24.б, където се наблюдава, че рамките за уеб разработка, базирани на един и същ език за програмиране, постигат сходни стойности, докато разликите с алтернативи, използващи различни езици, са по-значителни.

Забелязва се и разместване в класирането между алтернативите A4 и A5. При прилагане на класическия WSM метод, A5 (Laravel) изпреварва A4 (Flask) поради по-

доброто си представяне по критериите. При модифицираният WSM резултатите се коригират от нововъведения коефициент, който изразява по-голямото предпочитание към езика за програмиране Python, на който се базира Flask, за сметка на PHP, използван при Laravel.

Всичко това е показателно как ключовият критерий води до промяна на резултатите и отразява по-точно предпочитанията на ЛВР в крайния резултат.

4.3.4. Резултати от численото тестване на модификацията на метода на претегленото произведение

При Сценарий 2, предназначен за тестване на модифицирания метод на претегленото произведение, ЛВР е младши софтуерен инженер. Това обосновава промяна в тегловите коефициенти за важност на критериите, като се увеличава значимостта на критерия C3 – класацията на StackOverflow. Причината е, че платформата съдържа множество въпроси, отговори и готови решения, които са особено полезни за начинаещи разработчици. Наред с това, се променят и предпочитанията към използвания програмен език, което води до нови стойности на ψ_i^k. Примерно изчисление, базирано на различните предпочитания (вж. Таблица 4.10) за алтернатива A1-React се изпълнява както следва:

$$\psi_1^1 = \prod_{n=1}^{1} \sum_{r_n=1}^{3} p_{r_n}^1 v_{1,r_n} = 0{,}22 * 1 + 0{,}32 * 0 + 0{,}46 * 0 = 0{,}22$$

По същият начин са генерирани коефициентите за останалите алтернативи.

Матрицата за индивидуално вземане на решение, обобщаваща атрибутите за Сценарий 2, е представена на Фиг. 4.25.

ЛВР-2 (Младши софтуерен инженер)

Индивидуална матрица на алтернативите

КРИТЕРИИ		Теглови коефициенти за важност на критериите	Оценки за алтернативите / Коефициенти за коригиране на резултатите. По алтернативи					
			A1 React	A2 Vue.js	A3 Django	A4 Flask	A5 Laravel	A6 Symfony
			Коефициенти за коригиране на резултатите на алтернативите					
C1	Език за програмиране	-	0,22	0,22	0,32	0,32	0,46	0,46
			Оценки за алтернативите					
C2	GitHub класация	0,27	0,99	1,00	0,89	0,89	0,90	0,81
C3	StackOverflow класация	0,73	0,97	0,87	0,97	0,83	0,93	0,86

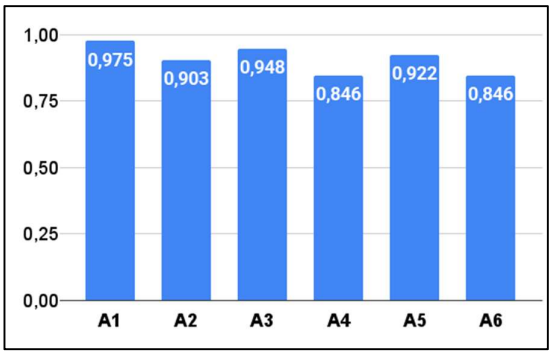
Фиг. 4.25. Матрица за индивидуално вземане на решение при Сценарий 2. Изглед от прототипът на система за подпомагане вземането на решения.

За всяка алтернатива данните от матрицата за вземане на решение, се агрегират чрез модифицирания математически модел на метода на претегленото произведение (2.30). Пример за изчисление на крайния резултат за алтернатива A1-React е представен по-долу.

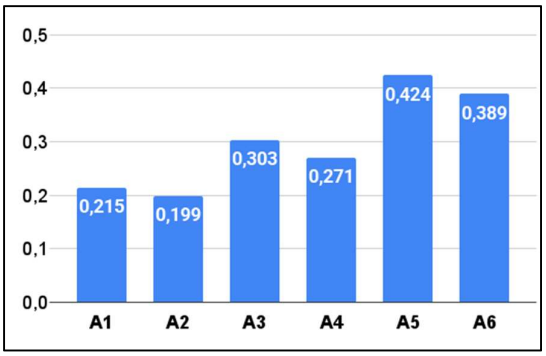
$$A_{1\,WPM}^{GDM} = \prod_{k=1}^{1} \psi_1^k \prod_{j=1}^{2} \left(a_{1,j}^k\right)^{w_j^k} = 0{,}22 * (0{,}99^{0{,}27} * 0{,}97^{0{,}73}) \approx 0{,}215$$

По аналогичен начин се агрегират резултатите за останалите алтернативи.

Отново, с цел да се определи влиянието на нововъведените коефициенти върху модифицирания WPM, са изчислени резултатите на алтернативите и по класическия метод на претегленото произведение (1.7). Сравнението между крайното класиране на алтернативите при двата метода е илюстрирано на Фиг. 4.26 (а, б).



а) Класически WPM



б) Модифициран WPM

Фиг. 4.26. Класиране на рамките за уеб разработка при Сценарий 2.

При класическия WPM най-предпочитана е алтернатива A1 (React), поради най-високата ѝ обща оценка по двата използвани критерия. В контраст, при модифицирания модел рамките, базирани на JavaScript, заемат последните позиции в класирането. Това се дължи на прилагането на коригиращия коефициент, който подобрява резултата на алтернативите, използващи езика PHP. В резултат, най-висока оценка получава A5 (Laravel) – рамка, базирана на PHP, която превъзхожда другата PHP-базирана алтернатива както по C2, така и по C3.

Отново се наблюдава сходство в резултатите между рамките, използващи един и същ програмен език (съответно със сходен коефициент), както и отчетливи различия при рамки, базирани на различни езици. Тези резултати потвърждават ефективността и приложимостта на модификацията и към метода на претегленото произведение.

4.3.5. Резултати от численото тестване на обобщения модел на метода на претеглената сума

Сценарий 3 има за цел да тества приложимостта на нововъведения, коефициент за предимство на алтернативите в комбинация с другите две модификации, които съставят така наречения в настоящия труд – обобщен модел на метода на претеглената сума.

Въз основа на входните данни от Таблици 4.9 – 4.11, в СПВР се генерират и предоставят формуляри за попълване от ЛВР, в които индивидуално да се изразят предпочитанията и да се определят оценки по критериите, за които е необходимо. Изглед от динамично генерираните и вече попълнени индивидуални формуляри за ЛВР-1, е представен на Фиг. 4.27.

ЛВР-1 (Старши софтуерен инженер)

Генериране на коефициенти за коригиране на резултатите на алтернативите (ψ_i^k)

Ключови критерии и опции		Коефи-циенти	Генерирани коефициенти					
			A1	A2	A3	A4	A5	A6
C1	Език за програмиране		0,46	0,46	0,33	0,33	0,21	0,21
O1	JavaScript	0,46						
O2	Python	0,33						
O3	PHP	0,21						

Генериране на оценки за алтернативите ($(b_{i,j}^k)^{norm}$)

Числено представени критерии		Cost	Benefit	Генерирани оценки					
				A1	A2	A3	A4	A5	A6
C2	GitHub рейтинг	○	●	0,990	1,000	0,890	0,890	0,900	0,810
C3	StackOverflow рейтинг	○	●	0,970	0,870	0,970	0,830	0,930	0,860

Генериране на оценки за алтернативите ($c_{i,j}^k$)

Матрично представени критерии		Оценка за опциите	Генерирани оценки					
			A1	A2	A3	A4	A5	A6
C4	Покритие на софтуерни слоеве		0,450	0,450	0,400	0,300	0,400	0,400
O5	Потребителски интерфейс	4						
O6	Клиентска логика	5						
O7	Сървърна логика	2						
O8	Работа с база данни	2						

Субективно определяне на оценки за алтернативите ($a_{i,j}^k$)

Субективно оценяеми критерии		Определяне на оценки					
		A1	A2	A3	A4	A5	A6
C5	Лекота за работа	0,850	0,750	0,700	0,800	0,650	0,540

Фиг. 4.27. Формуляри за изразяване на оценки и предпочитания, при обобщен модел. Изглед от прототипът на система за подпомагане вземането на решения за ЛВР-1.

За да се състави матрица за групово вземане на решение, следва ЛВР да определи:

- предпочитанието си за език за програмиране, в съответствие с ограничение (2.25), за да се генерират коефициенти за предимство на алтернативите ψ_i^k;
- предпочитанието си за минимизиране или максимизиране на стойностите на числено представените критерии при нормализация, за да се генерира оценката $(b_{i,j}^k)^{norm}$;
- предпочитанието си към опциите на матрично представените критерии, за да се генерира оценката $c_{i,j}^k$;
- субективните си оценки $a_{i,j}^k$, в съответствие с ограничение (2.29).

Същите формуляри се генерират и за другия участник в процеса по групово вземане на решения – ЛВР-2, както е показано на Фиг. 4.28.

ЛВР-2 (Младши софтуерен инженер)

Генериране на коефициенти за коригиране на резултатите на алтернативите (ψ_i^k)

Ключови критерии и опции		Коефи-циенти	Генерирани коефициенти					
			A1	A2	A3	A4	A5	A6
C1	Език за програмиране		0,22	0,22	0,32	0,32	0,46	0,46
O1	JavaScript	0,22						
O2	Python	0,32						
O3	PHP	0,46						

Генериране на оценки за алтернативите ($(b_{i,j}^k)^{norm}$)

Числено представени критерии		Cost	Benefit	Генерирани оценки					
				A1	A2	A3	A4	A5	A6
C2	GitHub класация	○	◉	0,990	1,000	0,890	0,890	0,900	0,810
C3	StackOverflow класация	○	◉	0,970	0,870	0,970	0,830	0,930	0,860

Генериране на оценки за алтернативите ($c_{i,j}^k$)

Матрично представени критерии		Оценка за опциите	Генерирани оценки					
			A1	A2	A3	A4	A5	A6
C4	Покритие на софтуерни слоеве		0,350	0,350	0,450	0,400	0,450	0,450
O5	Потребителски интерфейс	5						
O6	Клиентска логика	2						
O7	Сървърна логика	3						
O8	Работа с база данни	1						

Субективно определяне на оценки за алтернативите ($a_{i,j}^k$)

Субективно оценяеми критерии		Определяне на оценки					
		A1	A2	A3	A4	A5	A6
C5	Лекота за работа	0,300	0,320	0,550	0,450	0,950	0,850

Фиг. 4.28. Формуляри за изразяване на оценки и предпочитания, при обобщен модел. Изглед от прототипът на система за подпомагане вземането на решения за ЛВР-2.

Генерираните оценки и коефициенти са в съответствие с (2.17), (2.18) и (2.23) и съчетават събраните при дефинирането на задачата обективни данни с изразените предпочитания на ЛВР.

Данните от така попълнените формуляри, заедно с определените от ЛВР коефициенти за важност на критериите (вж. Таблица 4.9) и определените за тях коефициенти за компетентност (вж. Таблица 4.12), са структурирани в матрица за групово вземане на решение. Изглед на генерираната матрица от прототипа на СПВР е представен на Фиг. 4.29.

Администратор

Обобщена матрица на алтернативите

	КРИТЕРИИ	Коефициенти за важност на критериите		Коефициенти за компетентност на ЛВР		Оценки за алтернативите / Коефициенти за коригиране на резултатите											
						A1 React		A2 Vue.js		A3 Django		A4 Flask		A5 Laravel		A6 Symfony	
		ЛВР-1: Старши софт. инж.	ЛВР-2: Младши софт. инж.	ЛВР-1: Старши софт. инж.	ЛВР-2: Младши софт. инж.	ЛВР-1	ЛВР-2	ЛВР-1	ЛВР-2	ЛВР-1	ЛВР-2	ЛВР-1	ЛВР-2	ЛВР-1	ЛВР-2	ЛВР-1	ЛВР-2
						Коефициенти за коригиране на резултатите на алтернативите											
C1	Език за програмиране	-	-	-	-	0,46	0,22	0,46	0,22	0,33	0,32	0,33	0,32	0,21	0,46	0,21	0,46
						Оценки за алтернативите											
C2	GitHub класация	0,26	0,10	0,70	0,30	0,99	0,99	1,00	1,00	0,89	0,89	0,89	0,89	0,90	0,90	0,81	0,81
C3	StackOverflow класация	0,17	0,32	0,45	0,55	0,97	0,97	0,87	0,87	0,97	0,97	0,83	0,83	0,93	0,93	0,86	0,86
C4	Покритие на софтуерни слоеве	0,41	0,16	0,80	0,20	0,45	0,35	0,45	0,35	0,40	0,45	0,30	0,40	0,40	0,45	0,40	0,45
C5	Лекота за работа	0,16	0,42	0,50	0,50	0,85	0,30	0,75	0,32	0,70	0,55	0,80	0,45	0,65	0,95	0,54	0,85

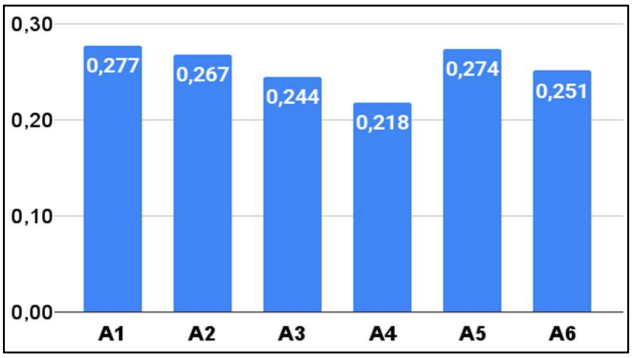
Фиг. 4.29. Матрица за групово вземане на решение при Сценарий 3. Изглед от прототипът на система за подпомагане вземането на решения.

Още при дефинирането на задачата, е избрано да се приложи линейна нормализация на стойностите на числено представените критерии, в съответствие с (1.12.а) и (1.12.б). Следващият пример демонстрира изчислението на агрегираната оценка за алтернатива A1 чрез обобщения модел на метода на претеглената сума (3.3).

$$A_{1\,WSM}^{GDM} = \sum_{k=1}^{2}\left(\left(\sum_{n=1}^{N}\sum_{r_n=1}^{R_n} p_{r_n}^{k} v_{1,r_n}\right)\sum_{j=1}^{4}\lambda_j^k w_j^k \beta_{1,j}^k\right) =$$

$$= 0{,}46 * (0{,}70 * 0{,}26 * 0{,}99 + 0{,}45 * 0{,}17 * 0{,}97 + 0{,}80 * 0{,}41 * 0{,}45 + 0{,}50 * 0{,}16 * 0{,}85) +$$
$$+ 0{,}22 * (0{,}30 * 0{,}10 * 0{,}99 \ + 0{,}55 * 0{,}32 * 0{,}97 + 0{,}20 * 0{,}16 * 0{,}35 + 0{,}50 * 0{,}42 * 0{,}30) = 0{,}277$$

По същия начин се изчисляват агрегираните оценки за останалите алтернативи, които се класират както е показано на Фиг. 4.30.



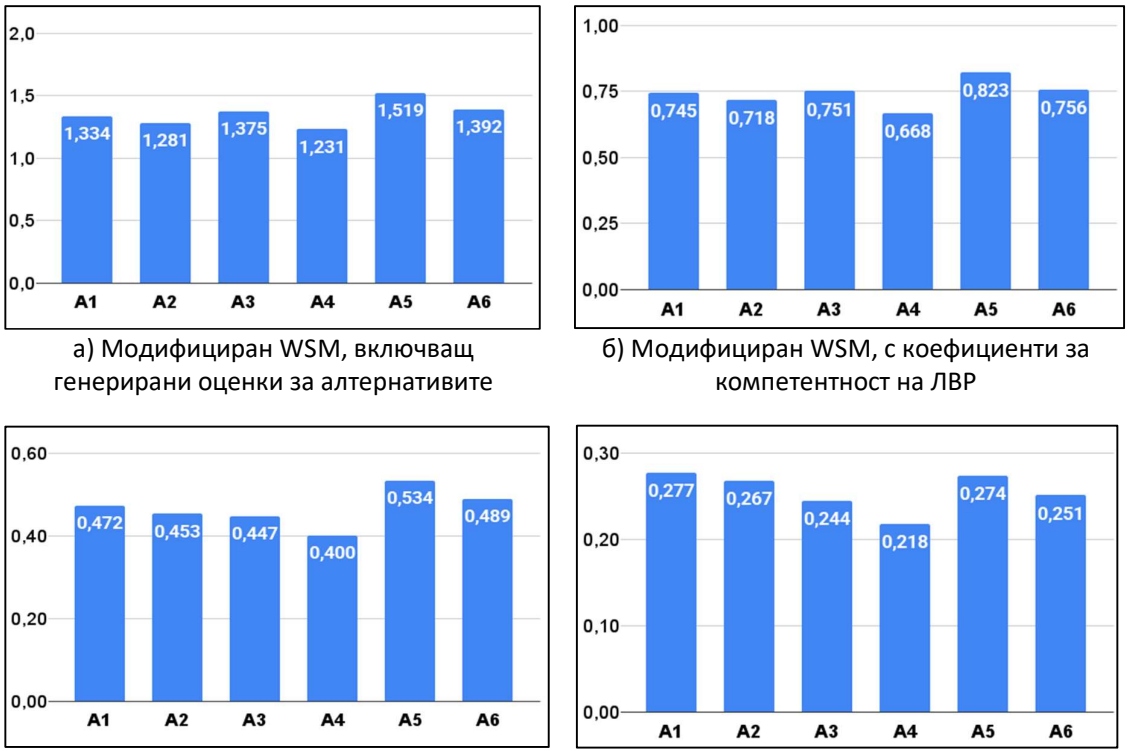
Фиг. 4.30. Класиране на рамките за уеб разработка при Сценарий 3.

При прилагане на обобщения алгоритъм, алтернатива A1-React постига най-висок резултат от 0,277, следвана плътно от A5-Laravel с резултат 0,274. Близките

стойности се дължат на ясно изразените предпочитания на двете ЛВР към противоположните програмни езици – JavaScript за ЛВР-1 и PHP за ЛВР-2. Разликата между зададените им коефициенти е само 0,01, което до голяма степен неутрализира ефекта на коригиращия фактор. Решаващо за крайната класация се оказва по-високата компетентност на ЛВР-1, който като старши софтуерен инженер притежава значително по-високи стойности на коефициентите за компетентност по критериите C2 и C4, както и сходни или равни стойности по C3 и C5 спрямо ЛВР2. Това води до доминиране на алтернативата, предпочетена от по-компетентния участник.

От това може да се заключи, че при групово вземане на решение, ефектът от коригиращите коефициенти може да се балансира от противоположни предпочитания между ЛВР. Въпреки това, включването на модификацията, отчитаща компетентността по критерии, позволява да се отрази по-голяма тежест на мнението на по-компетентните ЛВР, което в крайна сметка влияе на финалното решение.

Интересно е да се проследи натрупаният ефект от последователното прилагане на всяка от модификациите, представен на Фиг. 4.31.



а) Модифициран WSM, включващ генерирани оценки за алтернативите

б) Модифициран WSM, с коефициенти за компетентност на ЛВР

в) Модифициран WSM, с коефициенти за предимство на алтернативите

г) Обобщен WSM

Фиг. 4.31. Класиране на рамките за уеб разработка на различните етапи от прилагането на отделните модификации, при Сценарий 3.

От Фиг. 4.31 се вижда, че прилагайки различните предложени модификации, се определя различна най-предпочитана алтернатива, отчитайки специфичните особености на решаваната задача.

Комбинирането на всички предложени модификации в един обобщен алгоритъм е както възможно, така и успешно. Причината за това е, че всяка модификация засяга различни стъпки от алгоритъма и не въвежда математически ограничения, които да възпрепятстват прилагането на останалите. Както се потвърждава от проведените числени експерименти, обобщеният модел позволява едновременно: (1) вземане предвид на различията в компетентността на ЛВР спрямо конкретни критерии; (2) редуциране на субективния фактор чрез съчетаване обективни данни с индивидуалните предпочитания на ЛВР и (3) въвеждане на корекционни механизми в крайните резултати, базирани на критично важни характеристики на алтернативите.

ЗАКЛЮЧЕНИЕ – РЕЗЮМЕ НА ПОЛУЧЕНИТЕ РЕЗУЛТАТИ

Благодарение на цифровизацията, днес повече от всякога до сега, е възможно да се използва разнообразие от множество данни. За формирането на информирано решение е необходимо тези данни да се обработят по подходящ начин, за да може да получим структурирана информация, на база на която да се вземе най-подходящото решение. В тази връзка, настоящото изследване е насочено към методите и моделите за вземане на решение, вкл. формирането на групово решение.

Обекти на изследване в настоящия дисертационен труд са метода на претеглената сума, метода на претегленото произведение и софтуерните архитектури, с които те могат да бъдат интегрирани в система за подпомагане вземането на решение. Двата метода са едни от най-популярните и фундаментални представители на оценъчните методи в многокритериалното вземане на решения. С течение на времето те, са били развивани чрез различни модификации. Тяхната еволюция показва, че не са забравени статични методи, а динамично се адаптират към нови практически потребности. В резултат на това стават все по-гъвкави, широко приложими и подходящи да се интегрират в системи за подпомагане вземането на решение. Като софтуерен инструмент за рационално вземане на решение, СПВР следва да се проектира вземайки предвид функционалните изисквания наложени от алгоритмите и математическите модели. Софтуерната архитектура се определя от съчетаването на различни стилове и шаблони в съответствие с нефункционалните изисквания на организацията, като инфраструктурни възможности, технологичния опит на екипа, очакван мащаб на използване на системата и други.

В настоящия дисертационен труд са описани предложените модификации на метода на претеглената сума и метода на претегленото произведение. Първата модификация въвежда коефициенти за компетентност на ЛВР за всеки критерии поотделно и така прецизирана влиянието на ЛВР при групово вземане на решение. Втората модификация формализира генерирането на оценки като съчетава предпочитанията на ЛВР с обективни данни и така редуцира субективността и улеснява ЛВР при оценяването на алтернативите. Третата модификация въвежда генерирането на нов тип коефициенти, чрез които може да се постигне потискане или усилване на агрегираните резултати на алтернативите и по този начин се удовлетворяват критични

изисквания на задачата. Резултатите от проведените експерименти потвърждават, че всички предложени модификации могат да се комбинират успешно, за метода на претеглената сума, в един обобщен алгоритъм. Различните комбинации между модификациите позволяват резултатите да се агрегират по един от 13 различни модела – 9 с метода на претеглената сума и 4 с метода на претегленото произведение.

Предложени и проектирани са две различни архитектури на СПВР, удовлетворяващи противоречиви нефункционални изисквания на системата. Предложената трислойна архитектура, следва класическия трислоен архитектурен шаблон. Избрано е да се следва монолитен архитектурен стил със синхронно управляван поток на събитията. Подходяща е за организации, които разполагат със собствена сървърна инфраструктура. Предложената безсървърна архитектура използва архитектурния шаблон на разпределени функции с асинхронно управление на потока на събитията. Благодарение на това се постига висока мащабируемост, минимална необходимост от управление на инфраструктура, както и много висока адаптивност към динамични натоварвания. Всичко това прави безсървърната архитектура подходяща за организации, които разчитат на облачни услуги и трябва да обслужва голям брой потребители, например корпоративни или публични. Въз основа на предложените и проектирани архитектури, са разработени прототипи на СПВР, с които да се тества приложимостта както на предложените модификации, така и на самите архитектури. Прототипите предоставят централизирано управление на данни, достъпност от всяка точка, автоматизиране на рутинни операции и модул за линеен анализ на междинните резултати.

На база на получените резултати от проведеното тестване както на предложените модификации, така и на предложените архитектури за реализиране на моделите за подпомагане вземането на решения, може еднозначно да се установи тяхната практическа приложимост, чрез решаването на конкретни реални задачи.

Като бъдещо развитие на изследванията, свързани с модификациите на методите за вземане на решение се планират следните дейности: (1) търсене на решение за преодоляване на ограничението за прилагане на модификацията, формализираща генериране на оценки за алтернативите към метода на претегленото произведение; (2) комбиниране на всички модификации и за двата метода; (3)

адаптиране на предложените модификации към други подобни методи за вземане на решение.

Планираните дейности за бъдещото развитие на разработените прототипи на СПВР са свързани с: (1) разработването им като напълно функциониращо приложение с академични цели като експерименти, демонстрации и обучение; (2) реализиране или интегриране с външна услуга за надграждане на мрежовата сигурност с оторизация и отчетност (AAA, Authentication, Authorization, Accounting); (3) интегриране на агенти с изкуствен интелект като виртуален експерт и автоматизиране на част от административните задачи чрез него.

Резултатите от представеното изследване са докладвани на 4 международни конференции. По-голяма част от резултати са публикувани в общо 5 научни публикации, като 1 от тях е в международно научно списание. Всичките 5 публикации са реферирани и индексирани в световноизвестна база данни с научна информация Scopus, като 3 от тях са в издания с SJR.

ПРИНОСИ

Въз основа на получените резултати, могат да се формулират следните приноси:

1) Предложена е модификация на алгоритъма и математическите модели на метода на претеглената сума и метода на претегленото произведение в условията на групово вземане на решение. Модификацията отчита различните области на компетентност на ЛВР, чрез въвеждане на теглови коефициенти за компетентност на ЛВР спрямо всеки отделен критерий.
2) Предложена е модификация на алгоритъма за реализиране на метода на претеглената сума, формализираща генериране на оценки за алтернативите. Вместо субективно определяне на оценките се въвежда механизъм за съчетаване на обективни данни за алтернативите с предпочитанията на ЛВР, чрез декомпозиране на критерии до опции. Модификацията е валидна за групово и индивидуално вземане на решение.
3) Предложена е модификация на алгоритъма и математическите модели на метода на претеглената сума и на метода на претегленото произведение, формализираща генериране на коефициенти, даващи предимство в общото представяне на алтернативите. Коефициентите се генерират спрямо ключови критерии, използвайки същия механизъм за декомпозиране на критериите до опции и съчетаване на обективни данни за алтернативите с предпочитанията на ЛВР. Модификацията е валидна за групово и индивидуално вземане на решение.
4) Предложен е обобщен алгоритъм и комбинирани математически модели, съчетаващи предложените модификации на двата метода. Различните комбинации позволяват резултатите да се агрегират чрез 13 различни модела – 9 с метода на претеглената сума и 4 с метода на претегленото произведение.
5) Предложени са две софтуерни архитектури (трислойна и безсървърна) за реализиране на система за подпомагане вземането на решения, удовлетворяващи различни нефункционални изисквания. Двете архитектури демонстрират различни архитектурни стилове за реализиране на предложените обобщен алгоритъм и комбинираните математически модели на

модифицираните метода на претеглената сума и метода на претегленото произведение.

6) Разработени са два прототипа на система за подпомагане вземането на решения в съответствие с предложените софтуерни архитектури, като включват и модул за интерпретация на резултатите. Прототипите са използвани за числено тестване на предложените модификации на метода на претеглената сума и метода на претегленото произведение, като убедително демонстрират тяхната приложимост.

СПИСЪК НА ПУБЛИКАЦИИТЕ ПО ДИСЕРТАЦИОННИЯ ТРУД

1. **Dimitrova, Z.**, Borissova, D., & Dimitrov, V. (**2024, а**). Web application based on serverless architecture to support group decision-making by scoring models. In: *Proceedings of the 2024 5th International Conference on Communications, Information, Electronic and Energy Systems*, 365-369, E-ISBN: 979-8-3503-5286-3; USB ISBN: 979-8-3503-5285-6; PoD ISBN: 979-8-3503-5287-0. https://doi.org/10.1109/CIEES62939.2024.10811190 (Scopus)
2. **Dimitrova, Z.**, Borissova, D., & Dimitrov, V. (**2024, б**). A methodology for alternatives ranking by estimations forming based on values from criteria decomposition into options. *WSEAS Transactions on Business and Economics, 21*, 2136-2144, Print ISSN: 1109-9526, E-ISSN: 2224-2899. http://dx.doi.org/10.37394/23207.2024.21.176 (**SJR 2024 = 0.192, Q4**) (Scopus)
3. **Dimitrova, Z.**, Borissova, D., Mikhov, R., & Dimitrov, V. (**2023**). Group decision-making involving competence of experts in relation to evaluation criteria: Case study for e-commerce platform selection. In: *Modelling and Development of Intelligent Systems* (Simian, D., Stoica, L.F., eds.), 42–53. Communications in Computer and Information Science, 1761, Springer, Cham, Print ISBN: 978-3-031-27033-8; Online ISBN: 978-3-031-27034-5. https://doi.org/10.1007/978-3-031-27034-5_3 (**SJR 2023= 0.203 Q4**) (Scopus)
4. **Dimitrova Z.**, Borissova D., & Dimitrov V. (**2021**). Design of Web Application with Dynamic Generation of Forms for Group Decision-Making. In: *Computer Information Systems and Industrial Management* (Saeed K., Dvorsky J., eds.), 112-123. Lecture Notes in Computer Science, 12883, Springer, Cham, Print ISBN: 978-3-030-84339-7; Online ISBN: 978-3-030-84340-3. https://doi.org/10.1007/978-3-030-84340-3_9 (**SJR 2021= 0.407 Q2**) (Scopus)
5. Borissova, D., **Dimitrova, Z.**, Dimitrov, V., Yoshinov, R., Garvanova, M., & Garvanov, I. (**2021**). Multi-attribute decision-making model for ranking of web development frameworks. In: *25th International Conference on Circuits, Systems, Communications and Computers*, 3-8, E-ISBN: 978-1-6654-2749-4; (PoD) ISBN: 978-1-6654-2750-0. https://doi.org/10.1109/CSCC53858.2021.00009 (Scopus)

ДЕКЛАРАЦИЯ ЗА ОРИГИНАЛНОСТ НА РЕЗУЛТАТИТЕ

Долуподписаната *Зорница Агресимова Димитрова* декларирам, че настоящата дисертация съдържа оригинални резултати, получени при проведени от мен научни изследвания с подкрепата и съдействието на научния ми ръководител. Резултатите, които са получени, описани и/или публикувани от други учени, са надлежно и подробно цитирани в библиографията.

Настоящата дисертация не е прилагана за придобиване на научна степен в друго висше училище, университет или научен институт.

Подпис: .....

/Зорница Димитрова/

БИБЛИОГРАФИЯ

1. Adu, T.F., Mensah, L.D., Rockson, M.A.D. & Kemausuor, F. (2025). Decision support systems for waste-to-energy technologies: A systematic literature review of methods and future directions for sustainable implementation in Ghana, *Heliyon, 11*(3), 42353. https://doi.org/10.1016/j.heliyon.2025.e42353
2. Ahmad, K. & Naimi, S. (2023). Developing WSM-AHP method for adaptive construction project management. *International Journal on Technical and Physical Problems of Engineering, 15*(4), 160-170.
3. Alanazi, H.O., Abdullah, A.H. & Larbani, M. (2013). Dynamic Weighted Sum Multi-Criteria Decision Making: Mathematical Model. *International Journal of Mathematics and Statistics Invention, 1*(2), 16-18. https://ijmsi.org/Papers/Volume.1.Issue.2/B012016018.pdf
4. Amini, A., Abdollahi, A. & Hariri-Ardebili, M.A (2024). An automated machine-learning-assisted stochastic-fuzzy multi-criteria decision making tool: Addressing record-to-record variability in seismic design, *Applied Soft Computing, 154*, 111354. https://doi.org/10.1016/j.asoc.2024.111354
5. Arman, A., Arman, H., & Hadi-Vencheh, A. (2022). The Homogeneous MADM Methods: Is Trade-Off between Attributes Important? *Comput Intell Neurosci*. https://doi.org/10.1155/2022/8629986
6. Arnott, D. & Pervan, G. (2014). A Critical Analysis of Decision Support Systems Research Revisited: The Rise of Design Science. *Journal of Information Technology, 29*(4), 269-293. http://dx.doi.org/10.1057/jit.2014.16
7. Aruldoss, M., Lakshmi, T. M., & Venkatesan, V. P. (2013). A survey on multi criteria decision making methods and its applications. *American Journal of Information Systems, 1*(1), 31-43. http://dx.doi.org/10.12691/ajis-1-1-5
8. Aytekin, A. (2021). Comparative Analysis of the Normalization Techniques in the Context of MCDM Problems. *Decision Making Applications in Management and Engineering, 4*(2), 1-25. http://dx.doi.org/10.31181/dmame210402001a
9. Bączkiewicz, A., Wątróbski, J., Kizielewicz, B. & Sałabun, W. (2021). Towards Objectification of Multi-Criteria Assessments: a Comparative Study on MCDA Methods. *16th Conference on Computer Science and Intelligence Systems*, 417-425. http://dx.doi.org/10.15439/2021F61
10. Baker, D., Bridges, D., Hunter, R., Johnson, G., Krupa, J., Murphy, J. & Sorenson, K. (2001). Guidebook to Decision-Making Methods, Department of Energy, USA.
11. Barkadehi, M., Nilashi, M., Ibrahim, O., Fardi, A. & Samad, S. (2018). Authentication Systems: A Literature Review and Classification. *Telematics and Informatics, 35*(5), 1491-1511, ISSN 0736-5853. https://doi.org/10.1016/j.tele.2018.03.018
12. Bass, L., Clements, P., Kazman, R. (2022). Software Architecture in Practice (4th ed.). *Software Engineering Institute Series in Software Engineering, Carnegie Mellon University*, *Addison-Wesley*, ISBN-10: 0-13-688609-4.
13. Behera, L., & Vardhan, V. (2025). End-to-End Data Pipelines: Redefining the Architecture of Data Engineering in Cloud Environments. *ESP International Journal of Advancements in Computational Technology, 2*(4), 26-33. http://dx.doi.org/10.56472/25839233/IJAST-V2I4P104

14. Belton, V., & Pictet, J. (1997). A Framework for Group Decision Using a MCDA Model: Sharing, Aggregating or Comparing Individual Information?. *Journal of Decision Systems, 6*(3), 283-303. http://dx.doi.org/10.1080/12460125.1997.10511726
15. Belton, V., & Stewart, T. J. (2002). Multiple criteria decision analysis: An integrated approach. *Boston, MA: Kluwer Academic Publishers*. https://doi.org/10.1007/978-1-4615-1495-4
16. Borissova, D. (2016). Group decision making for selection of k-best alternatives. *Comptes rendus de l'Academie bulgare des Sciences, ISSN 1310-1331, 69*(2), 183-190. https://www.iict.bas.bg/ipdss/DBorissova/2016-Compt-Rendus-69-2-83-190.pdf
17. Borissova, D., Korsemov, D. & Mustakerov, I. (2018). Multi-attribute group decision making considering difference in experts knowledge: An Excel application. *In Proc. 12th Int. Management Conference – Management Perspectives in the Digital Era*, 387-395. https://conference.management.ase.ro/archives/2018/pdf/3_1.pdf
18. Bozorg-Haddad, O., Loáiciga, H., & Zolghadr-Asli, B. (2021). Simple Weighting Methods: Weighted Sum and Weighted Product Methods. *In: A handbook on Multi-Attribute Decision-Making Methods. Wiley & Sons Publication Inc.*, 17-24. https://doi.org/10.1002/9781119563501
19. Briggs-Myers, I., & Myers, P. B. (1995). Gifts differing: understanding personality type. *Mountain View, CA, Davies-Black Publishing*, ISBN 0-89106-074-X.
20. Buschman, F., Meunier, R., Rohnert, H., Sommerlad, P., Stal, M. (1996). Pattern-Oriented Software Architecture. A System of Patterns. *John Wiley & Sons.*
21. Cervantes, H., & Kazman, R. (2024). Designing Software Architectures (2nd ed.). *Software Engineering Institute Series in Software Engineering, Carnegie Mellon University*, ISBN-10: 0-13-810802-1.
22. Cinelli, M., Kadzinski, M., Gonzalez, M., & Słowiński, R. (2020). How to Support the Application of Multiple Criteria Decision Analysis? Let Us Start with a Comprehensive Taxonomy, *Omega, 96*(2), 102261. http://dx.doi.org/10.1016/j.omega.2020.102261
23. Cinelli, M., Kadzinski, M., Miebs, G., Gonzalez, M., & Słowiński, R. (2022). Recommending Multiple Criteria Decision Analysis Methods with A New Taxonomy-based Decision Support System. *European Journal of Operational Research, 302*(2). http://dx.doi.org/10.1016/j.ejor.2022.01.011
24. Chakraborty, S., Raut, R.D., Rofin, T.M. & Chakraborty S. (2023). A comprehensive and systematic review of multi-criteria decision-making methods and applications in healthcare. *Healthcare Analytics, 4*, 100232. https://doi.org/10.1016/j.health.2023.100232
25. Chang, S., Li, A., Wang, X., & Wang, X. (2022). Joint optimization of e-commerce supply chain financing strategy and channel contract. *European Journal of Operational Research, 303*(2), 908–927. https://doi.org/10.1016/j.ejor.2022.03.013
26. Chen, M., Mao, S. & Liu, Y. (2014). Big Data: A Survey. *Mobile Networks and Applications, 19*, 171–209. https://doi.org/10.1007/s11036-013-0489-0
27. Chen, Y., Jin, Q., Fang, H., Lei, H., Hu, J., Wu, Y., Chen, J., Wang, C., Wan, Y. (2019). Analytic network process: Academic insights and perspectives analysis. *Journal of Cleaner Production, 235*, 1276-1294. https://doi.org/10.1016/j.jclepro.2019.07.016
28. Chourabi, Z., Khedher, F., Dhouib, A. & Cheikhrouhou, M. (2018). Multi-criteria decision making in workforce choice using AHP, WSM and WPM. *The Journal of The Textile Institute, 110*, 1-10. http://dx.doi.org/10.1080/00405000.2018.1541434

29. Churchman, C. W., & Ackoff, R. L. (1954). An approximate measure of value. *Journal of the Operations Research Society of America, 2*(2), 172–187. https://doi.org/10.1287/opre.2.2.172
30. Clements, P., Bachmann, F., Bass, L., Garlan, D., Ivers, J., Little, R., Merson, P., Nord, R., & Stafford, J. (2010). Documenting Software Architectures: Views and Beyond. *Second Edition. Boston: Addison-Wesley*, ISBN 978-0-321-55268-6.
31. Dean, M. (2022). A Practical Guide to Multi-criteria Analysis. *Technical Report, Bartlett School of Planning, University College London*. http://dx.doi.org/10.13140/RG.2.2.15007.02722
32. Decancq, K., & Lugo, M. A. (2009). Setting Weights in Multidimensional Indices of Well-being and Deprivation. *Oxford Poverty & Human Development Initiative, 18*. https://ophi.org.uk/sites/default/files/OPHI-wp18.pdf
33. De Keyser, W., Peeters, P. (1996). A note on the use of PROMETHEE multicriteria methods. *European Journal of Operational Research, 89*(3), 457-461. https://doi.org/10.1016/0377-2217(94)00307-6
34. Djartov, B., Oswald, F. & Jakobi, J. (2024). Through the Psychological Lens: Unveiling Biases in Multi-Criteria Decision-Making. *In: Tareq Ahram, Luca Casarotto and Pietro Costa (eds) Human Interaction and Emerging Technologies, 1*. http://doi.org/10.54941/ahfe1005476
35. Dyer, J., Figueira, J., Greco, S., & Ehrogott, M. (2005). MAUT — Multiattribute Utility Theory. *In book: Multiple Criteria Decision Analysis: State of the Art Surveys. Springer.* http://dx.doi.org/10.1007/0-387-23081-5_7
36. Eckenrode, R. T. (1965). Weighting Multiple Criteria. *Management Science, 12*, No. 3, 180-192.
37. Edwards, W. (1992). Utility theories: measurements and applications. https://doi.org/10.1007/978-94-011-2952-7
38. Ehrgott, M., Köksalan, M., Kadziński, M. & Kalyanmoy D. (2025). Fifty years of multi-objective optimization and decision-making: From mathematical programming to evolutionary computation. *European Journal of Operational Research*. https://doi.org/10.1016/j.ejor.2025.06.012
39. Engau, A. (2010). Interactive Decomposition-Coordination Methods for Complex Decision Problems. *In: Zopounidis, C., Pardalos, P. (eds) Handbook of Multicriteria Analysis. Applied Optimization, Springer, 103,* 329-365. https://doi.org/10.1007/978-3-540-92828-7_12
40. Engert, M., Evers, J., Hein, A., & Krcmar, H. (2022). The engagement of complementors and the role of platform boundary resources in e-Commerce platform ecosystems. *Information Systems Frontiers, 24*(6), 2007–2025. https://doi.org/10.1007/s10796-021-10236-3
41. Esangbedo, M. O. & Che, A. (2016). Grey Weighted Sum Model for Evaluating Business Environment in West Africa. *Mathematical Problems in Engineering.* https://doi.org/10.1155/2016/3824350
42. Fattoruso, G. (2022). Multi-Criteria Decision Making in Production Fields: A Structured Content Analysis and Implications for Practice. *Journal of Risk and Financial Management, 15*(10), 431. https://doi.org/10.3390/jrfm15100431
43. Ferretti, V. (2016). From stakeholders analysis to cognitive mapping and Multi-Attribute Value Theory: An integrated approach for policy support. *European Journal of Operational Research, 253*(2), 524-541. https://doi.org/10.1016/j.ejor.2016.02.054
44. Figueira, J.R., Greco, S., Roy, B. & Słowinski, R. (2010). ELECTRE Methods: Main Features and Recent Developments. *In: Zopounidis, C., Pardalos, P. (eds) Handbook of Multicriteria Analysis. Applied Optimization, Springer, 103, 51*-89. http://dx.doi.org/10.1007/978-3-540-92828-7

45. Fishburn, P. C. (1967). Additive Utilities with Incomplete Product Set: Applications to Priorities and Assignments, *Operations Research.*
46. Fishburn, P.C. (1970). Utility theory for decision making. (No. RACR105). https://apps.dtic.mil/sti/tr/pdf/AD0708563.pdf
47. Foote, B., & Yoder, J. (1997). Big Ball of Mud. *Department of Computer Science University of Illinois at Urbana-Champaign, Urbana, IL, USA*. https://www.plopcon.org/pastplops/plop97/Proceedings/foote.pdf
48. Fostervold, K. I., Aarås, A., & Lie, I. (2006). Work with visual display units: Long-term health effects of high and downward line-of-sight in ordinary office environments. *International Journal of Industrial Ergonomics, 36*(4), 331–343. https://doi.org/10.1016/j.ergon.2005.05.003
49. Fowler, M., & Lewis, J. (2014). Microservices: a definition of this new architectural style. https://martinfowler.com/articles/microservices.html?utm_source=chatgpt.com
50. Gallagher, K. M., Cameron, L., Boulé, M., & De Carvalho, D. (2019). Does using multiple computer monitors affect health and productivity? A systematic review. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 63*(1), 977. https://doi.org/10.1177/1071181319631210
51. Galster, M., Eberlein, A., & Moussavi, M. (2010). Systematic selection of software architecture styles. *Software, IET, 4*(5), 349-360. http://dx.doi.org/10.1049/iet-sen.2009.0004
52. Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). Design Patterns: Elements of Reusable Object-Oriented Software. *Addison-Wesley Professional: Boston, MA, USA*.
53. Gardziejczyk, W. & Zabicki, P. (2017). Normalization and variant assessment methods in selection of road alignment variants – case study. *Journal of Civil Engineering and Management, 23*, 510-523. http://dx.doi.org/10.1016/j.procs.2022.01.156
54. Garlan, D., & Shaw, M. (1994). An Introduction to Software Architecture. *School of Computer Science Carnegie Mellon University Pittsburgh, PA.* https://userweb.cs.txstate.edu/~rp31/papers/intro_softarch.pdf
55. Giove, S., Brancia, A., Satterstrom, F.K., & Linkov, I. (2009). Decision support systems and environment: Role of MCDA. *In: Marcomini, A., Suter II, G., Critto, A. (eds) Decision Support Systems for Risk-Based Management of Contaminated Sitesp*, 1–21. https://doi.org/10.1007/978-0-387-09722-0_3
56. Greco, S., Ehrgott, M. & Figueira, J. R. (2016). Multiple Criteria Decision Analysis - State of the Art Surveys. *Springer*. https://doi.org/10.1007/978-1-4939-3094-4
57. Greco, S., Słowiński, R. & Wallenius, J. (2025). Fifty years of multiple criteria decision analysis: From classical methods to robust ordinal regression. *European Journal of Operational Research, 323*(2), 351-377. https://doi.org/10.1016/j.ejor.2024.07.038
58. Goswami, S.S., Behera, D.K., & Mitra, D.S. (2020). A comprehensive study of weighted product model for selecting the best product in our daily life. *Brazilian Journal of Operations & Production Management, 17*(2), 1–18. https://doi.org/10.14488/BJOPM.2020.017
59. Hassan, M. A., Shukur, Z., & Hasan, M. K. (2020). An efficient secure electronic payment system for E-Commerce. *Computers, 9*(3), 66. https://doi.org/10.3390/computers9030066
60. Hoffman, A. (2024). Web Application Security, 2nd Edition. *O'Reilly*, 444. https://www.oreilly.com/library/view/web-application-security/9781098143923/
61. Hwang, C.L. & Lin, M.J. (1987). Group decision making under multiple criteria. *Springer-Verlag 1, Berlin.*Dang, V.L., Wan, S. & Guo, J. (2025). Third-party logistics outsourcing: A review of two decades of advancing decision-making approaches with an up-to-date three-layer criteria

framework integrating environmental, social, and governance metrics. *International Journal of Production Economics, 284*, 109615. https://doi.org/10.1016/j.ijpe.2025.109615

62. Hwang, C.L. & Yoon, K. (1981). Multiple attribute decision making: Methods and applications. A state-of-the-art survey. *Springer-Verlag 1, New York*.
63. Jafaryeganeh, H., Ventura, M. & Soares, C. G. (2020). Effect of normalization techniques in multi-criteria decision making methods for the design of ship internal layout from a Pareto optimal set. *Structural and Multidisciplinary Optimization, 62*(3). https://link.springer.com/article/10.1007/s00158-020-02581-9
64. Jansen, S.J.T. (2011). The multi-attribute utility method. *In: Jansen, S., Coolen, H., Goetgeluk, R. (eds) The Measurement and Analysis of Housing Preference and Choice,* 101–125. https://doi.org/10.1007/978-90-481-8894-9_5
65. Jiang, L., Pei, Y., & Zhao, J. (2020). Overview Of Serverless Architecture Research. *Journal of Physics: Conference Series. 1453*(1), 012119. http://dx.doi.org/10.1088/1742-6596/1453/1/012119
66. Kabak, Ö., & Ervural, K. (2017). Multiple attribute group decision making: A generic conceptual framework and a classification scheme. *Knowledge-Based Systems, 123*, 13-30. http://dx.doi.org/10.1016/j.knosys.2017.02.011
67. Kaur, P. & Kumar, S. (2013). An intuitionistic fuzzy simple additive weighting (IFSAW) method for selection of vendor. *IOSR Journal of Business and Management, 15*(2), 78-81. https://www.iosrjournals.org/iosr-jbm/papers/Vol15-issue2/K01527881.pdf
68. Kawasaki, T., Wakashima, H., & Shibasaki, R. (2021). The use of e-commerce and the COVID-19 outbreak: A panel data analysis in Japan. *Transport Policy, 115*, 88–100. https://doi.org/10.1016/j.tranpol.2021.10.023
69. Keeney, R.L. (1996). Value-focused thinking: Identifying decision opportunities and creating alternatives. *European Journal of Operational Research, 92*(3), 537-549. https://doi.org/10.1016/0377-2217(96)00004-5
70. Keeney, R. L., & Gregory, R. S. (2005). Selecting attributes to measure the achievement of objectives. *Operations Research, 53*(1), 1–11. https://doi.org/10.1287/opre.1040.0158
71. Keeney, R.L., & Raiffa, H. (1993). Decisions with multiple objectives. *Cambridge University Presss, Cambridge, MA.* https://doi.org/10.1017/CBO9781139174084
72. Khan, Sh., Chaabane & Dweiri, A.F. (2018). Multi-Criteria Decision-Making Methods Application in Supply Chain Management: A Systematic Literature Review. *Multi-Criteria Methods and Techniques Applied to Supply Chain Management Publisher: Intechopen.* http://dx.doi.org/10.5772/intechopen.74067
73. Kouraklis, J. (2016). MVVM as Design Pattern. *In book: MVVM in Delphi*. http://dx.doi.org/10.1007/978-1-4842-2214-0_1
74. Kumar, A. (2014). Software Architecture Styles: A Survey. *International Journal of Computer Applications, 87*(9). http://dx.doi.org/10.5120/15234-3768
75. Laaziri, M., Benmoussa, K., Khoulji, S., & Kerkeb, M.L. (2019). A comparative study of PHP frameworks performance. *Procedia Manufacturing, (32)*, 2019, 864-871. https://doi.org/10.1016/j.promfg.2019.02.295
76. Liao, H., He, Y., Wu, X., Wu, Z. & Bausys, R. (2023). Reimagining Multi-Criterion Decision Making by Data-Driven Methods Based on Machine Learning: A Literature Review. *Information Fusion, 100*. https://doi.org/10.1016/j.inffus.2023.101970

77. Lopez, L.M., Ishizaka, A., Qin, J., & Alvarez Carrillo, P.A. (2023). Chapter 1 - Multi-Criteria Decision-Making. *Multi-Criteria Decision-Making Sorting Methods, Academic Press*, 1-11. https://doi.org/10.1016/B978-0-32-385231-9.00006-7
78. MacCrimmon, K. R. (1968). Decisionmaking among multiple-attribute alternatives: A survey and consolidated approach. *ARPA, RAND Corporation*. https://www.rand.org/content/dam/rand/pubs/research_memoranda/2009/RM4823.pdf
79. Magabaleh, A., Ghraibeh, L., Audeh, A., Albahri, A. S., Deveci, M. & Antucheviciene, J. (2024). Systematic review of software engineering uses of multi-criteria decision-making methods: Trends, bibliographic analysis, challenges, recommendations, and future directions. *Applied Soft Computing, 163*, 111859. https://doi.org/10.1016/j.asoc.2024.111859
80. Malefaki, S., Markatos, D., Filippatos, A., & Pantelakis, S. (2025). A Comparative Analysis of Multi-Criteria Decision-Making Methods and Normalization Techniques in Holistic Sustainability Assessment for Engineering Applications. *Aerospace, 12*(2), 100. https://doi.org/10.3390/aerospace12020100
81. Malczewski, J. (1999). GIS and multicriteria decision analysis. *J. Wiley & Sons, New York*.
82. Mardani, A., Jusoh, A., MD Nor, K., Khalifah, Z., Zakwan, N., & Valipour, A. (2015). Multiple criteria decision-making techniques and their applications – a review of the literature from 2000 to 2014. *Economic Research-Ekonomska Istraživanja, 28*(1), 516–571. https://doi.org/10.1080/1331677X.2015.1075139
83. Marmsoler, D. (2014). Towards a theory of architectural styles. *In: 22nd ACM SIGSOFT International Symposium*, 823-825. http://dx.doi.org/10.1145/2635868.2661683
84. Mateo, J. (2012). Weighted Sum Method and Weighted Product Method. *In book: Multi Criteria Analysis in the Renewable Energy Industry*, Springer, 19-22. http://dx.doi.org/10.1007/978-1-4471-2346-0_4
85. McClelland, D. C. (1973). Testing for competence rather than for "intelligence." *American Psychologist, 28*(1), 1–14. https://doi.org/10.1037/h0034092
86. Mohammadi, M. & Rezaie, J. (2023). Ratio product model: A rank-preserving normalization-agnostic multi-criteria decision-making method. *Journal of Multi-Criteria Decision Analysis*, 163-172. http://dx.doi.org/10.1002/mcda.1806
87. Mokrushin, V. O. & Ilinykh, S. P. (2024). Designing Web Application Client Program Modules for Asynchronous Event Handling. *3rd International Conference on Problems of Informatics, Electronics and Radio Engineering (PIERE)*, Novosibirsk, 720-723. https://doi.org/10.1109/PIERE62470.2024.10805021
88. Mustakerov, I. & Borissova, D. (2014). A Web application for group decision-making based on combinatorial optimization. *In Proc. of 4th International Conference on Information Systems and Technologies, 46-56.*
89. Nasr, A., Johansson, J., Ivanov, O., Björnsson, I. & Honfi, D. (2022). Risk-based multi-criteria decision analysis method for considering the effects of climate change on bridges. *Structure and Infrastructure Engineering, 19*(13). http://dx.doi.org/10.1080/15732479.2022.2033278
90. Niu, B., Dong, J., Dai, Z., & Liu, Y. (2021). Sales data sharing to improve product development efficiency in cross-border e-commerce. *Electronic Commerce Research and Applications, 51*, 101112. https://doi.org/10.1016/j.elerap.2021.101112
91. Opricovic, S., Tzeng, G.-H. (2004). Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS. *European Journal of Operational Research, 156*(2), 445-455. https://doi.org/10.1016/S0377-2217(03)00020-1

92. Perry, D. E., & Wolf, A. L. (1992). Foundations for the study of software architecture. *ACM SIGSOFT Software Engineering Notes, 17*(4), 40. https://doi.org/10.1145%2F141874.141884
93. Petković, D. L., Madić, M. J., & Mitković, M. M. (2025). Development of a Decision Support System for Biomaterial Selection Based on MCDM Methods. *Applied Sciences, 15*(16), 9198. https://doi.org/10.3390/app15169198
94. Pramanik, P. K. D., Biswas, S., Pal, S., Marinković, D., & Choudhury, P. (2021). A Comparative Analysis of Multi-Criteria Decision-Making Methods for Resource Selection in Mobile Crowd Computing. *Symmetry, 13*(9), 1713. https://doi.org/10.3390/sym13091713
95. Peng, Y. & Shi, Y. (2012). Editorial: Multiple criteria decision making and operations research. *Annals of Operations Research, 197*, 1–4. https://doi.org/10.1007/s10479-012-1104-7
96. Podvezko, V. (2011). The comparative analysis of MCDA methods SAW and COPRAS. *Engineering Economics, 22*. 134-146. http://dx.doi.org/10.5755/j01.ee.22.2.310
97. Razmak, J. & Aouni, B. (2014). Decision Support System and Multi-Criteria Decision Aid: A State of the Art and Perspectives. *Journal of Multi-Criteria Decision Analysis, 22*(1-2). http://dx.doi.org/10.1002/mcda.1530
98. Ribeiro, R. A., Moreira, A. M., Van Den Broek, P. M., & Pimentel, A. (2011). Hybrid Assessment Method for Software Engineering Decisions. *Decision support systems, 51*(1), 208-219. https://doi.org/10.1016/j.dss.2010.12.009
99. Richards, M. (2015). The Challenges of Service-Based Architecture. *No Fluff Just Stuff (NFJS)*. https://developertoarchitect.com/articles/sba-challenges.pdf
100. Richards, M., & Ford, N. (2020). Fundamentals of Software Architecture. An Engineering Approach. *O'Reilly Media*.
101. Rodríguez-Carrillo, M.L., Pérez-Domínguez, L., Romero-López, R., Luviano-Cruz, D. & León-Castro, E. (2025). A systematic literature review on the use of multicriteria decision making methods for small and medium-sized enterprises innovation assessment. *Front. Artif. Intell. Sec. AI in Business (8)*1605756. https://doi.org/10.3389/frai.2025.1605756
102. Roy, B. (1991). The outranking approach and the foundations of electre methods. *Theory and Decision, 31*, 49–73. https://doi.org/10.1007/BF00134132
103. Saaty, R.W. (1987). The analytic hierarchy process—what it is and how it is used. *Mathematical Modelling, 9*(3–5), 161-176. https://doi.org/10.1016/0270-0255(87)90473-8
104. Sadowski, C., Storey, MA., & Feldt, R. (2019). A Software Development Productivity Framework. *In: Sadowski, C., Zimmermann, T. (eds) Rethinking Productivity in Software Engineering. Apress*, Berkeley, CA, 39-47. https://doi.org/10.1007/978-1-4842-4221-6_5
105. Saleh, A. M., & Hilal, M. A. (2024). Identification Risks in Road Construction Projects Using Weighted Product Model: Iraq as a Case Study. *Journal of Engineering, 30*(12), 76–95. https://doi.org/10.31026/j.eng.2024.12.06
106. Salih, M.M., Zaidan, B.B., Zaidan, A.A., Ahmed, M.A. (2019). Survey on fuzzy TOPSIS state-of-the-art between 2007 and 2017. *Computers & Operations Research, 104,* 207-227. https://doi.org/10.1016/j.cor.2018.12.019
107. Sharma, A., Kumar, M., & Agarwal, S. (2015). A Complete Survey on Software Architectural Styles and Patterns. *Procedia Computer Science, 70*(2), 16-28. http://dx.doi.org/10.1016/j.procs.2015.10.019
108. Sharot, T., Velasquez, C. M., & Dolan, R. J. (2010). Do Decisions Shape Preference? Evidence From Blind Choice. *Psychological Science, 21*(9), 1231-1235. https://doi.org/10.1177/0956797610379235

109. Shethiya, A. S. (2025). Scalability and Performance Optimization in Web Application Development. *Integrated Journal of Science and Technology*, 2(1). https://ijstpublication.com/index.php/ijst/article/view/1
110. Shiloh, S., Koren, S., & Zakay, D. (2001). Individual differences in compensatory decision-making style and need for closure as correlates of subjective decision complexity and difficulty. *Personality and Individual Differences, 30*(4), 699-710. https://doi.org/10.1016/S0191-8869(00)00073-8
111. Singh, A. & Malik, S. K. (2014). Major MCDM Techniques and their application – A Review. *IOSR Journal of Engineering, 04*(05), 15-25. https://iosrjen.org/Papers/vol4_issue5%20(part-2)/C04521525.pdf
112. Sorooshian, S. & Parsia, Y. (2019). Modified Weighted Sum Method for Decisions with Altered Sources of Information. *Mathematics and Statistics, 7*(3), 57-60. http://dx.doi.org/10.13189/ms.2019.070301
113. Sotoudeh-Anvari, A. (2022). The applications of MCDM methods in COVID-19 pandemic: A state of the art review. *Applied Soft Computing, 126*, 109238. https://doi.org/10.1016/j.asoc.2022.109238
114. Spencer, L.M., & Spencer, S.M. (1993). Competence at Work: Models for Superior Performance. *John Wiley & Sons Press*, New York, ISBN: 978-0-471-54809-6.
115. Stanujkic, D., Popovic, G., Karabasevic, D., Meidute-Kavaliauskiene, I. & Ulutas, A. (2021). An Integrated Simple Weighted Sum Product Method—WISP. *IEEE Transactions on Engineering Management, 99.* 1-12. http://dx.doi.org/10.1109/TEM.2021.3075783
116. Taherdoost, H. (2023). Analysis of Simple Additive Weighting Method (SAW) as a MultiAttribute Decision-Making Technique: A Step-by-Step Guide. *Journal of Management Science & Engineering Research, (6)*1, 21-24. http://dx.doi.org/10.30564/jmser.v6i1.5400
117. Taherdoost, H., & Madanchian, M. (2023). Multi-Criteria Decision Making (MCDM) Methods and Concepts. *Encyclopedia, 3*(1), 77-87. https://doi.org/10.3390/encyclopedia3010006
118. Talukder, B., Blay-Palmer, A., Hipel, K. W., & VanLoon, G. W. (2017). Elimination Method of Multi-Criteria Decision Analysis (MCDA): A Simple Methodological Approach for Assessing Agricultural Sustainability. *Sustainability, 9*(2), 287. https://doi.org/10.3390/su9020287
119. Tekinerdogan, B., & Verdouw, C. (2020). Systems Architecture Design Pattern Catalog for Developing Digital Twins. *Sensors, 20*(18), 5103. https://doi.org/10.3390/s20185103
120. Tan, T., Mills, G., Papadonikolaki, E. & Liu, Z. (2021). Combining multi-criteria decision making (MCDM) methods with building information modelling (BIM): A review, *Automation in Construction, Volume 121*, 103451 https://doi.org/10.1016/j.autcon.2020.103451
121. Triantaphyllou, E. (2000) Multi-Criteria Decision Making Methods. *In: Multi-criteria Decision Making Methods: A Comparative Study. Applied Optimization, Springer, Boston, MA, (44)*, 320. https://doi.org/10.1007/978-1-4757-3157-6_2
122. Triantaphyllou, E., & Mann, S. H. (1989). An examination of the effectiveness of multi-dimensional decision-making methods: A decision-making paradox. *Decision Support Systems, 5*(3), 303–312. https://doi.org/10.1016/0167-9236(89)90037-7
123. Triantaphyllou, E. & Sánchez, A. (1997). A Sensitivity Analysis Approach for Some Deterministic Multi-Criteria Decision-Making Methods. *Decision Sciences. 28*(1). 151 - 194. http://dx.doi.org/10.1111/j.1540-5915.1997.tb01306.x

124. Triantaphyllou, E. Shu, B., Sanchez, S. & Ray, T. (1998). Multi-criteria decision making: An operations research approach. *Encyclopedia of Electrical and Electronics Engineering, 15*, 175-186.
125. Tofallis, C. (2014). Add or Multiply? A Tutorial on Ranking and Choosing with Multiple Criteria. *INFORMS Transactions on Education, 14*(3), 109-119. https://doi.org/10.1287/ited.2013.0124
126. Tofallis, C. (2022). Objective Weights for Scoring: The Automatic Democratic Method. *Multiple Criteria Decision Making (17)*, 69-84. https://doi.org/10.22367/mcdm.2022.17.04
127. Tu, Z. (2023). Research on the Application of Layered Architecture in Computer Software Development. *Journal of Computing and Electronic Information Management, 11*(3), 34-38. http://dx.doi.org/10.54097/jceim.v11i3.08
128. Turban, E., Sharda, R., & Delen, D. (2011). Decision support and business intelligence systems (9th ed.). *Pearson Education*.
129. UK Government Analysis Function. (2024). An Introductory Guide to Multi-Criteria Decision Analysis (MCDA). https://analysisfunction.civilservice.gov.uk/policy-store/an-introductory-guide-to-mcda/
130. Vafaei, N., Ribeiro, R. & Camarinha-Matos, L. (2022). Assessing Normalization Techniques for Simple Additive Weighting Method. *Procedia Computer Science, 199*, 1229-1236. http://dx.doi.org/10.1016/j.procs.2022.01.156
131. Velasquez, M. & Hester, P.T. (2013). An analysis of multi-criteria decision making methods. *International Journal of Operations Research, 10*(2), 56-66.
132. Wagner, S., & Deissenboeck, F. (2019). Defining Productivity in Software Engineering. *In: Sadowski, C., Zimmermann, T. (eds) Rethinking Productivity in Software Engineering. Apress*, Berkeley, CA, 29-38. http://dx.doi.org/10.1007/978-1-4842-4221-6_4
133. Wang, Z., Nabavi, S. R., & Rangaiah, G. P. (2024). Multi-Criteria Decision Making in Chemical and Process Engineering: Methods, Progress, and Potential. *Processes, 12*(11), 2532. https://doi.org/10.3390/pr12112532
134. Wawrzoniak, M., Bruno, R., Klimovic, A., & Alonso G. (2024). Boxer: FaaSt Ephemeral Elasticity for Off-the-Shelf Cloud Applications. *ETH Zurich Research Collection*. https://doi.org/10.3929/ethz-b-000712599
135. Williams, A. & Cai, Y. (2025). Insights into Weighted Sum Sampling Approaches for Multi-Criteria Decision Making Problems. *WSEAS TRANSACTIONS ON MATHEMATICS. 24*. 327-346. http://dx.doi.org/10.37394/23206.2025.24.32
136. Wu, X., Zhu, X., Wu, G. -Q., & Ding, W. (2014). Data mining with big data. *In IEEE Transactions on Knowledge and Data Engineering, (26)*1, 97-107. https://doi.org/10.1109/TKDE.2013.109
137. Yudhistira, N. A., Setiawansyah, N., Ardiansah, N. T., Maryana, N. S., Yadin, N. Y., & Oktaviani, N. R. (2024). Development of Multi-Attribute Utility theory methods in dynamic decision models using Change-Data driven. *Evergreen, 11*(4), 3279–3289. https://doi.org/10.5109/7326962
138. Zaidan, A., Muzammil, H., Ahmed A-H., Kiah, M., Laiha, M. & Mohamed, A. (2015). Multi-Criteria Analysis for OS-EMR Software Selection Problem: A Comparative Study. *Decision Support Systems. 78*(3), 15-27. http://dx.doi.org/10.1016/j.dss.2015.07.002
139. Zanakis, S. H., Solomon, A., Wishart, N. & Dublish, S. (1998). Multi-attribute decision making: A simulation comparison of select methods. European Journal of Operational Research, 107(3), 507-529, ISSN 0377-221. https://doi.org/10.1016/S0377-2217(97)00147-1

140. Zavadskas, E. K. & Turskis, Z. (2011). Multiple criteria decision making (MCDM) methods in economics: an overview. *Technological and Economic Development of Economy, 17*(2), 397–427. https://doi.org/10.3846/20294913.2011.593291
141. Zavadskas, E. K., Turskis, Z., Antucheviciene, J., & Zakarevicius, A. (2012). Optimization of weighted aggregated sum product assessment. *Elektronika Ir Elektrotechnika, 122*(6), 3-6. https://doi.org/10.5755/j01.eee.122.6.1810
142. Zayat, W., Kilic, H. S., Yalcin, A.S., Zaim, S. & Delen D. (2023). Application of MADM methods in Industry 4.0: A literature review. *Computers & Industrial Engineering, 177*, 109075. https://doi.org/10.1016/j.cie.2023.109075
143. Zheng, Y., Liang H., Hao Ch., Fan L. (2008). Computer Vision Syndrome: A widely spreading but largely unknown epidemic among computer users. *Computers in Human Behavior, (24)*5, 2026-2042, ISSN 0747-5632. https://doi.org/10.1016/j.chb.2007.09.004