DEX2Web – A Web-Based Software Implementing the Multiple-Criteria Decision-Making Method DEX[1]

Adem Kikaj[1, 2] and Marko Bohanec[2]

[1] Jožef Stefan International Postgraduate School, Jamova 39, Ljubljana 1000, Slovenia
[2] Jožef Stefan Institute, Department of Knowledge Technologies, Jamova 39, Ljubljana 1000, Slovenia
{adem.kikaj,marko.bohanec}@ijs.si

Abstract. DEX2Web is an online suite of tools to help individuals and groups with their decision-making. DEX2Web implements the qualitative multiple-criteria decision-modelling method DEX. DEX is useful for supporting complex decision-making tasks, where there is a need to select a particular option from a set of possible ones to satisfy the goals of the decision-maker. DEX2Web primarily supports interactive development and evaluation of DEX models. Most of the functionality of the first available version of DEX2Web is inherited from its desktop ancestor DEXi: development of DEX model structure, editing of attributes and their scales, definition of decision rules, multi-attribute evaluation and analysis of alternatives, and presenting evaluation results with charts. DEX2Web has a modern software architecture and employs a newly developed DEX software library. DEX2Web is freely available on http://dex2web.ijs.si/.

Keywords: DEX2Web, DEX, decision-making software, multiple-criteria, web, software, group decision-making

1 Introduction

Decision-making is the process of identifying and choosing alternatives based on the values, preferences, and beliefs of the decision-maker. Some decision problems are inherently difficult because of various obstacles, such as missing information, uncertainty, conflicting goals, and opposing views of multiple decision-makers. In such situations, decision-making may substantially benefit from using decision-aiding methods and tools.

Decision problems of a type that involve multiple, possibly conflicting criteria, can be aided using *Multiple-Criteria Decision-Making (MCDM)* methods [1, 2]. The aim is to help the decision maker understand better and structure a decision problem to represent it in the form of a decision model and use this model for decision-making tasks,

[1] This is the final authors' version of the article, presented at ICDSST 2021 (https://icdsst2021.wordpress.com/). The final authenticated version is available online https://doi.org/10.1007/978-3-030-73976-8_3.

such as choosing, ranking/sorting decision alternatives, and analyzing the gained results [2].

To support the use of MCDM methods in practice, various multi-criteria *decision-making software (DMS)* have been developed [3, 4]. Decision-making software usually provides ways to build a decision model and analyze the results. Decision-making software consists of various forms of computer programs designed to enable users to process a set of goals to be achieved, alternatives available for making them, and relations between goals and options. *World Wide Web (WWW)* technologies have rapidly transformed the design, development, and implementation of all types of DMS [5, 6]. In this work, we are focused on the decision-making method DEX and its implementation as a web-based DMS.

Decision Expert (DEX) is a qualitative MCDM method. Currently, the DEX method is implemented in freely available software called *DEXi* [10]. DEXi is useful for supporting complex decision-making tasks, and it runs as a desktop application on the Microsoft Windows platforms. DEXi supports two primary functions: (1) development of qualitative multi-attribute models and (2) applying these models for the evaluation and analysis of decision options. DEX software is also distributed partly in other computing platforms. DEXi has been extensively and successfully used in numerous national and international projects [7].

DEXi software was released 20 years ago. Even though it has been regularly updated and extended with new features, it has reached the state that calls for a thorough renovation. In this work, we are designing modern web-based software called *DEX2Web* with an enhanced architecture that integrates the DEX method through the newly developed *DEX library*.

The structure of this paper is as follows. The next two sections describe the DEX method and supporting software, respectively. The implementation of DEX2Web is presented in Section 4. In Section 5, we conclude this work.

2 Qualitative Multiple-Criteria Method DEX

DEX [7, 8, 9, 11, 12] is a qualitative, hierarchical, multi-criteria decision-modelling method for the evaluation and analysis of decision alternatives. DEX decision models have a hierarchical structure, representing a decomposition of some decision problems into smaller, less complex sub-problems. DEX models are developed by defining (i) attributes, (ii) attribute scales, (iii) hierarchically structure of the attributes, and (iv) decision rules [13].

The DEX model thus consists of:

- *Attributes*: variables that represent basic features and assessed values of decision alternatives. DEX models can have one or more *root* attribute(s), i.e., those that do not have any *parent* attributes. Attributes that do not have any *child* attributes are called *input* attributes; all other attributes are *aggregated* attributes.
- *Scales* of attributes: these are *qualitative* scales and consist of a set of words, such as: “excellent”, “acceptable”, “inappropriate”, etc. Usually, but not necessarily, scales are *ordered preferentially*, i.e., from bad to good values.

- *Hierarchy* of attributes: represents the decomposition of the decision problem and relations between attributes; higher-level attributes depend on lower-level ones. In general, a hierarchy is a directed graph without cycles.
- *Aggregation* function: to aggregate qualitative values, DEX primarily uses decision tables [13], which can be interpreted as collections of *if-then* rules. To define an aggregation function, the decision-maker defines an output value for each combination of input attribute's values.

Decision alternatives (also called *options* in DEXi) are defined by qualitative values, which are assigned to the model's input attributes. DEX evaluates alternatives in a bottom-up way, progressively aggregating the values according to the hierarchical structure of the model. Values of aggregated attributes, for which their respective children attributes already have values assigned, are aggregated using the corresponding aggregating functions. The final evaluation of an alternative is represented by the values calculated at the model's roots.

The DEX model that will be illustrated in this section will be used throughout this work to describe different properties of the DEX method. The model described here is the well-known DEX model *Car* [14], which is distributed together with the DEXi software.

Table 1. On the left side, the structure of the attributes is presented, and on the right side, their corresponding scales are displayed. The scales are ordered from worst values (shown in red) to best values (shown in green).

Attribute	**Scale**
CAR	unacc, acc, good, exc
├─PRICE	high, medium, low
│ ├─BUY.PRICE	high, medium, low
│ └─MAINT.PRICE	high, medium, low
└─TECH.CHAR.	bad, acc, good, exc
├─COMFORT	small, medium, high
│ ├─#PERS	to_2, 3-4, more,
│ ├─#DOORS	2, 3, 4, more
│ └─LUGGAGE	small, medium, big
└─SAFETY	small, medium, high

The Car model is a simple model for evaluating cars, where the input attributes influence only one parent attribute. All the attributes have qualitative values assigned, which is shown in Table 1. There are 10 attributes where 6 are *input*, and 4 are *aggregated*, including one *root* attribute.

Table 2 shows the *aggregation function* (decision rules) of the *CAR* attribute. The two leftmost columns correspond to the two attributes that influence *CAR*: *PRICE* and *TECH.CHAR*. These two columns contain all possible combinations of their input values. The third column gives the *output* value of the aggregation function for the respective row. The third column is filled-in by the decision-maker.

Hereafter, this paper is focused on the implementation of the DEX2Web software. For further methodological issues and recommendations for DEX model development, for instance, how to choose and structure attributes and how to define value scales, the reader is referred to [7, 8, 9, 10].

Table 2. Decision table of the aggregated *CAR* attribute.

	PRICE	*TECH.CHAR.*	***CAR***
1	high	bad	**unacc**
2	high	acc	**unacc**
3	high	good	**unacc**
4	high	exc	**unacc**
5	medium	bad	**unacc**
6	medium	acc	**acc**
7	medium	good	**good**
8	medium	exc	**exc**
9	low	bad	**unacc**
10	low	acc	**good**
11	low	good	**exc**
12	low	exc	**exc**

3 Software for DEX

Three main generations of DEX qualitative modelling computer programs have been developed so far:

1. *DECMAK* [15] was released in 1981 for operating systems RT-11, VAX/VMS, and later for MS-DOS.
2. *DEX* [16] was released in 1987 as an integrated interactive computer program for MS-DOS.
3. *DEXi* [10] was released in 2000 as an interactive educational program for MS-Windows.

Over the years, additional features were added to DEXi, which gradually became a complete, stable and *de-facto* standard implementation of the DEX method. DEXi supports an interactive creation and editing of all components of DEX models (attributes, their hierarchy and scales, decision tables and alternatives), and provides methods for the evaluation and analysis of alternatives (what-if analysis, “plus-minus-1” analysis, selective explanation, comparison of alternatives). DEXi is free software, available at http://kt.ijs.si/MarkoBohanec/dexi.html. There are other implementations related to DEX, for instance, *proDEX* [17], a stand-alone Python implementation of DEX, and DEXx [9], a Java-based library.

DEX library is a new software library, developed from scratch, that implements all the DEX features proposed through decades of development in a unified, compact and modern software architecture. The library's main components have been designed according to the proposal of the so-called *Extended DEX* [9] and include:

- *DexProjects* is a new class aimed at managing multiple DEX models within the same decision project.
- *DexModels* represents a single DEX model.
- *DexAttributes* represents a single attribute (variable) of a DEX model. Attributes can be structured into trees or true hierarchies (directed acyclic graphs).
- *DexScales* represents attribute value scales. In addition to qualitative scales, which are exclusively used in DEXi, the new library also supports bounded and unbounded integer and continuous scales.
- *DexValues* is a class that represents different values that can be assigned to attributes. In addition to qualitative values used in DEXi, the DEX library supports numerical values, intervals, sets, probabilistic and fuzzy distributions, and offsets.
- *DexFunctions* represents aggregation functions. In comparison with DEXi, a number of new function types have been introduced, including constant, weighted, marginal, discretization and programmable functions.
- *DexAlternatives* is a component that represents decision alternatives.
- *DexEvaluation* provides the functionality needed to evaluate the existing alternatives of a DEX model, using various value propagation methods: single value, interval, set, or probabilistic or fuzzy value distribution.
- *DexViews* represents different states of objects of classes of the library that need to be represented in the user interface. The output of a *DexView* can be a JSON or XML object.
- *DexEditors* contain a collection of classes for editing components of DEX models: projects, models, scales, aggregation functions and alternatives.

The DEX library is used as the core component of DEX2Web.

4 DEX2Web

The main purpose of DEX2Web is to provide a new modern software supporting the DEX method. DEX2Web employs the DEX library and provides a web-based user interface so that the software can be used through an internet browser. In the current version, which is presented in this paper, we primarily implemented the functionality that is currently available in DEXi. The web-based environment also encouraged us to add some *group decision-making* features.

In this section, we describe the functional and non-functional requirements for the software, followed by a description of software architecture and an example of using the system.

4.1 Functional Requirements

The functional requirements of the DEX2Web are defined as follows:

- *Usage of the DEX2Web* – DEX2Web allows the users to create or import a single DEX project, which may contain multiple DEX models. DEX projects are stored for

registered users. DEX2Web is backwards compatible with DEXi and can import models created by DEXi.

- *Registration* – This function allows the users of DEX2Web to register/create an account through a web browser. At this level, the input data required are in harmony with those required from the OAuth [18] internet protocol.
- *Log in* – This function allows users to log into the DEX2Web to use the functionality provided for registered users in the system.
- *Account editor* – This function allows users to edit all their provided account data given at the registration level.
- *Project editor* – This function allows DEX2Web users to create, edit, delete, import, export, and share a DEX project with other users. A DEX project can be saved on the DEX2Web server or on a local machine by exporting. An owner of a DEX project can add an unlimited number of users as members of that project.
- *User roles* – This function allows the users to specify a specific user's role in the case when they share a specific DEX project. Currently supported roles are *Decision Maker* and *Decision Analyst*.
- *Model editor* – This function allows users to create, modify, delete, import, export a DEX model, and share it with other users. The models supported in the current version have the same components as in DEXi, i.e., they are restricted to qualitative attributes and rule-based aggregation functions.
- *Attributes* – This function allows users to create, modify, and delete attributes. The concept of linked attributes in DEXi [10] is replaced by the more general concept of using full attribute hierarchies [9]. No restrictions are imposed upon the number of attributes included in a model. The largest DEX models constructed so far did not exceed 500 attributes [7], and this number is easily manageable by DEX2Web.
- *Scales* – This function allows users to create and modify scales of attributes. This version of DEX2Web supports categorical scales with any number of values, the same as DEXi.
- *Aggregation functions* – This function allows users to define and modify aggregation functions of attributes of type aggregated. Aggregation functions are currently represented only in the form of decision tables, the same as in DEXi. For practical reasons, the size of the tables is by default limited to the maximum of 2000 entries.
- *Values* – This function allows users to define DEX values to scales of attributes, the output of decision rules, and decision alternatives. The supported values are analogous to those in DEXi.
- *Alternatives* – This function allows users to define, modify, and delete alternatives.
- *Evaluation* – This function allows users to evaluate the defined alternatives. In this version of DEX2Web, only qualitative evaluation is supported.
- *Improved user-interface* – This function allows the users to see a DEXi-like view where one attribute from the model is selected, but also extends it with displays of properties and details of attributes, which are selectable by the user.
- *Editor of multiple DEX models* – This function allows users to edit up to five different DEX models simultaneously.
- *Search* – This function allows users to search in terms of DEX projects and DEX models by their names.

- *Charts* – This function allows users to build interactive charts. The charts supported by DEX2Web are of the type *Bar* and *Radar*.
- *Group decision-making* – This function provides support for group decision-making in two ways; (1) while sharing a DEX project with multiple users and (2) sharing a DEX model based on the Web Socket protocol [19]. Each user can define any number of models within a given project and share them with other users. There is no prescribed limit of the number of users sharing the same DEX model.

4.2 Non-Functional Requirements

The non-functional requirements of DEX2Web are:

- *Usability*
  - The user interface is implemented using Thymeleaf.
  - The styling of web pages is done by Bootstrap front-end open-source toolkit.
  - Changes of the web pages' style are done in the Bootstrap toolkit's source files, and they are recompiled each time any change appears.
- *Reliability/Availability*
  - The software is deployed on a server that meets minimum requirements for a Spring project of version 2.3.4 that runs with Java 8.
  - The platform is designed to be available at all times, except in the rare cases of maintenance.
  - Before any update or upgrade of the platform, users are informed in advance through email and a notification on the platform's main page.
- *Scalability*
  - The whole platform project is built as a single JAR (Java ARchive).
  - The database is physically separated from the web-server.
  - The platform media are saved within the server.
  - The extension of volumes in storage space is done physically.
- *Performance*
  - The client requests are managed by thread connection pool.
  - The production database connections are managed by HikariCP [21] that supports the connection pool.
  - The port is 1000 Mbit/s with unlimited traffic.
  - The base of each webpage is light, and it downloads it under 250ms.
  - Small AJAX requests are supported and allowed. The same is applied for web socket connections.
  - DEX2Web supports up to 1000 threads simultaneously.
- *Serviceability*
  - A specific documentation tool is developed within this project called *DexDoc*. This tool provides documentation of the platform DEX2Web and the DEX library. The DexDoc is interactive and can handle the next versions of both components.
  - The platform's logging is done using *Apache Commons Logging*, and for the DEX library, it is done from a built-in logging mechanism within the library.

  - The *Aspect-Oriented Programming (AOP)* is used to increase the modularity of cross-cutting concerns.
- *Security*
  - The server where DEX2Web is deployed is secured using network restrictions.
  - The web session timeout is set to 30 minutes.
  - HTTPS protocol is used to communicate between the browser and the server.
  - The validation of imported DEX files is done on two sides, user and server-side.
  - Nothing in the database is permanently deleted.
  - Database backup is done automatically every day.
  - Database backups are relocated manually to another secure server each week.
  - The authorization and authentication of the user are taken care of by following Spring Security [20].

4.3 Multi-Layered Architecture

DEX2Web implements the DEX method using the newly developed DEX library. To fulfil the functional and non-functional requirements defined above, and to reduce the development time, we used a Java framework known as Spring Boot [20].

The architecture of DEX2Web is multi-layered due to the web-based nature of software. The software architecture is dependent on the architecture of the Spring Boot framework.

DEX2Web is a large-scale web-based enterprise application. DEX2Web application consists of two main parts: (1) resources and (2) business logic. The first group of resources consists of view layouts and media type resources such as images (158 files). The second group of DEX2Web application consists of Java classes (78 Java files) structured mainly following the *model-view-controller* design pattern.

DEX2Web implements a 4-layer architecture (see Fig. 1) consisting of:

- *Presentation layer*: Web pages for handling the dialogue with the user (decision maker) and invoking the necessary services.
- *Business layer*: It consists of the DEX library that provides means and methods for the creation and modification of DEX models, and evaluation and analysis of decision alternatives. Also, it includes all the other components that are mainly supported by the Spring Boot framework, such as handling the MVC design pattern and providing utilities needed to implement functional requirements.
- *Persistence layer*: It consists of components where we set up the communication channel with the database following *data-access-object* and *data-transfer-object* patterns.
- *Database layer*: A relational database designed using *MySQL*, which is used to store data and to facilitate sufficient information sharing between users and the library to provide full functionality of DEX2Web.

In order to reduce risks associated with updates and changes of external software components, we have tried to use the least number of well-established, mature, free and open-source components, supported by a large community.

This implementation of DEX2Web is the first available online DMS that supports the DEX method for model development, and evaluation and analysis of alternatives. The software can be used remotely, which opens up new possibilities in decision support. To support group decision-making, such DMS requires a network to share information, and the Internet serves for this purpose. DEX2Web supports both individual and group decision-making.

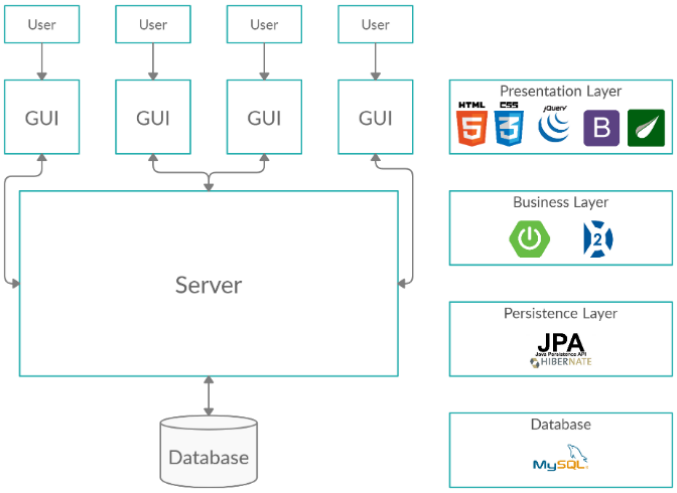

Fig. 1. Multi-layered architecture of DEX2Web.

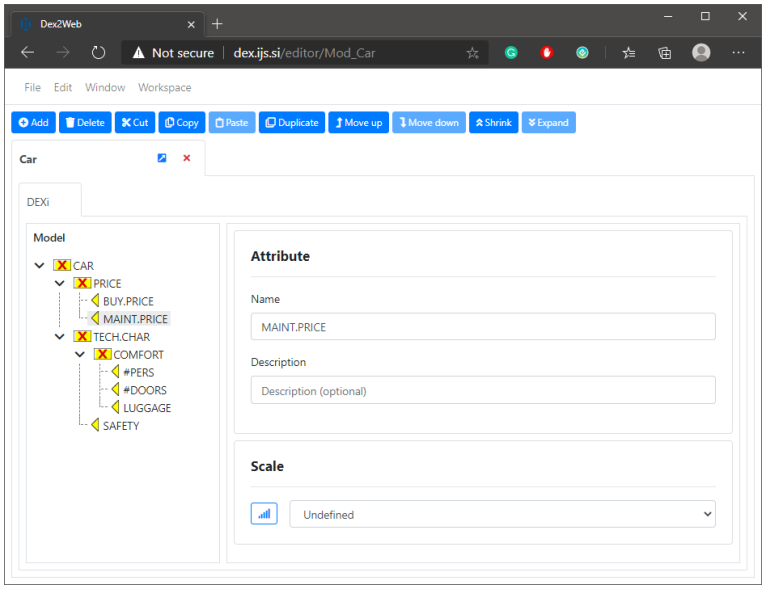

Fig. 2. Tree Structure of the Car model in DEX2Web.

4.4 Example of Using DEX2Web

This section presents a DEX2Web use-case, illustrating the development of DEX models and evaluation of decision alternatives. The example uses the CAR model, presented in section 2.

Fig. 2 shows the model editor of DEX2Web, which provides functionality for defining the hierarchical structure of the model, naming, and describing attributes, and accessing editors of other DEX model components (scales and aggregation functions). Fig. 2 shows the structure of the CAR model in the stage where scales of attributes and aggregation functions have not been defined yet.

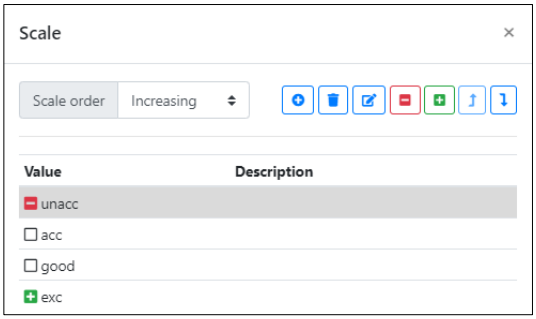

Fig. 3. Defining scale of the *CAR* attribute in DEX2Web.

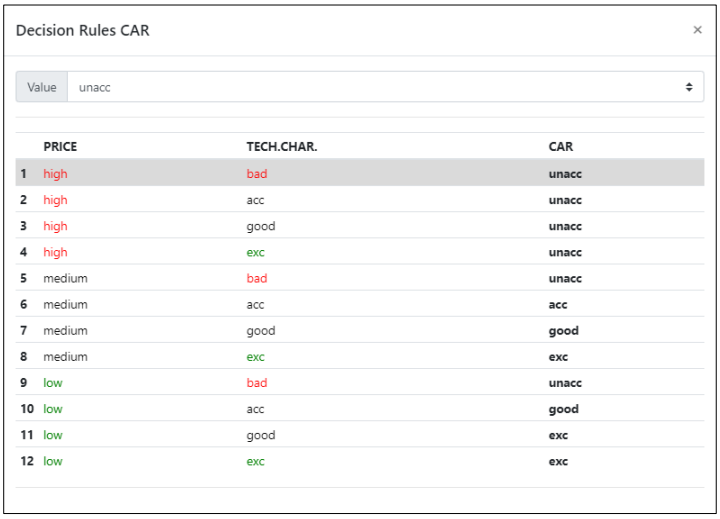

Fig. 4. Aggregation function of the *CAR* attribute of the Car model in DEX2Web.

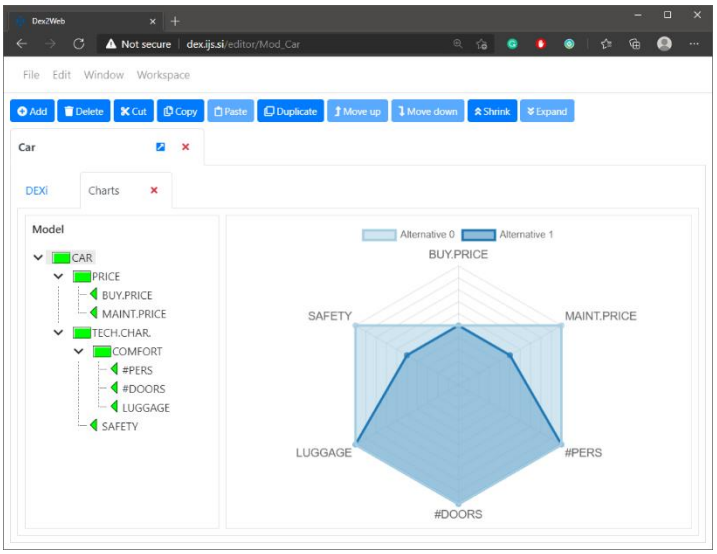

Fig. 5. Evaluated results of alternatives shown using radar charts in DEX2Web.

Fig. 3 shows the window used to define the scale of an attribute – specifically, the scale of the root attribute *CAR*, which consists of four preferentially-ordered qualitative values: *unacc*, *acc*, *good*, and *exc*.

The next step is to define the aggregation function of all aggregated attributes. Fig. 4 shows the function editor while editing the decision table from Table 2. By default, output values are assigned to the value "*" (asterisk) that represents the set of all values of the scale of that attribute. Fig. 4 shows the defined aggregation function of *CAR* aggregated attribute where the decision-maker has already defined the output values in the right-most column.

Table 3. Functionality comparison between DEXi and DEX2Web.

Functionality	**DEXi**	**DEX2Web**
Online	✗	✓
DEX projects	✗	✓
Tabular view of DEX models	✗	✓
DEX attributes	✓	✓
DEX scales	✓	✓
DEX aggregation functions	✓	✓
Scale order in aggregation functions	✓	✗
Weight in aggregation functions	✓	✗
Reports	✓	✗
Editor of multiple DEX models	✓	✓
Multiple view creator	✗	✓
Interactive charts	✗	✓
Group decision-making	✗	✓

After defining all model components, the decision-maker can define decision alternatives and evaluate them. Fig. 5 shows the results of two evaluated alternatives (cars) using radar charts.

Table 3 presents a comparison of supported functionality of DEXi and DEX2Web. The latter is the first online DEX implementation and extends DEXi by supporting DEX projects (as collections of DEX models), provides additional interactive views of DEX models and charts, and supports some basic group decision-making features (sharing of models). At this stage, some advanced DEXi features (considering scale orders and attribute weights to aid aggregation function editing, advanced reports) are not implemented in DEX2Web, but will be gradually introduced in the future.

5 Conclusions

The purpose of this work was to design a new decision-making software that implements the DEX method in a web-based environment. The main goals were to define a modern software architecture and user interface, suitable for the web, and to utilize the newly developed DEX library, which is meant to provide an extended set of methods and tools for the new generation of DEX software.

DEX2Web is the first fairly complete implementation of the DEX method available for interactive use on the Internet (http://dex2web.ijs.si/). DEX2Web is backwards compatible with DEXi: it can import, but not export, DEXi models; the latter is due to the extended functionality of the DEX library, which is not available in DEXi. The first version of DEX2Web was completed in October 2020, so it is still a brand-new software without much-gained experience about its use in practice.

Currently, DEX2Web mainly resembles the functionality that is already available in the desktop software DEXi. In addition to user and safety management functions, which are necessary for the web environment, DEX2Web currently provides only minor extensions to DEXi: multiple models collected within DEX projects extended model-editing views and model sharing. However, the architecture of DEX2Web is flexible, and together with the new DEX library, additional features will be gradually added to DEX2Web. These include combining qualitative and quantitative attributes, using extended value types in aggregation functions and evaluation of alternatives (intervals, fuzzy and probability value distribution, and samples), multiple types of aggregation functions (e.g., for conversion between qualitative and quantitative attributes and weighted aggregation using weights), and additional support for group decision-making.

Acknowledgment

The authors acknowledge Jožef Stefan International Postgraduate School, where this work has been conducted, and the Public Scholarship, Development, Disability and Maintenance Fund of the Republic of Slovenia (Contract no. 11011-44/2017-14) that financially supported this research. The authors also acknowledge the financial support from the Slovenian Research Agency (research core funding No. P2-0103).

References

1. Bouyssou, Denis, et al. Evaluation and Decision Models with Multiple Criteria: Stepping Stones for the Analyst. Vol. 86. Springer Science & Business Media, 2006.
2. Greco, Salvatore, Jose Figueira, and Matthias Ehrgott. Multiple Criteria Decision Analysis. New York: Springer, 2016.
3. Ishizaka, Alessio, and Philippe Nemery. Multi-criteria Decision Analysis: Methods and Software. John Wiley & Sons, 2013.
4. Mustajoki, Jyri, and Mika Marttunen. Comparison of Multi-Criteria Decision Analytical Software-Searching for Ideas for Developing a New EIA-Specific Multi-Criteria Software. 2013.
5. Bhargava, Hemant K., Daniel J. Power, and Daewon Sun. Progress in Web-Based Decision Support Technologies. Decision Support Systems 43.4, 2007: 1083-1095.
6. Power, Daniel J. Web-Based and Model-Driven Decision Support Systems: Concepts and Issues. AMCIS 2000 Proceedings, 2000: 387.
7. Bohanec, Marko. Multi-Criteria DEX Models: An Overview and Analysis. The 14th international symposium on operational research in Slovenia, eds. LZ Stirn, MK Borštar, J. Žerovnik and S. Drobne (Ljubljana: Slovenian Society Informatika–Section for Operational Research), 2017.
8. Bohanec, Marko, et al. DEX Methodology: Three Decades of Qualitative Multi-Attribute Modeling. Informatica 37.1, 2013.
9. Trdin, Nejc, and Marko Bohanec. Extending the Multi-Criteria Decision Making Method DEX with Numeric Attributes, Value Distributions and Relational Models. Central European Journal of Operations Research 26.1, 2018: 1-41.
10. Bohanec, Marko. DEXi: Program for Multi-Attribute Decision Making, User's Manual, Version 5.04. IJS Report DP-13100, Ljubljana, Slovenia: Institut Jozef Stefan, 2020.
11. Bohanec, Marko, and Vladislav Rajkovič. DEX: An Expert System Shell for Decision Support. Sistemica 1.1, 1990: 145-157.
12. Kikaj, Adem, and Marko Bohanec. Towards Web-based Decision Modeling Software based on DEX Methodology. Central European Conference on Information and Intelligent Systems. Faculty of Organization and Informatics Varazdin, 2019.
13. Kikaj, Adem, and Bohanec, Marko. Complex Decision Rules in DEX Methodology: jRule Algorithm and Performance Analysis. Proceedings of the 21th International Conference Information Society IS 2018, Volume A, Ljubljana, 17-20, 2018.
14. Bohanec, Marko, and B. Zupan. Car Evaluation Dataset. UCI Machine Learning Repository, Accessed 25.11.2020. https://archive.ics.uci.edu/ml/datasets/car+evaluation.
15. Bohanec, Marko, and Vladislav Rajkovič. DECMAK: An Expert System Shell for Multi-Attribute Decision Making. Institut Jožef Stefan, 1988.
16. Šet, Andrej, Marko Bohanec, and Marjan Krisper. "VREDANA: Program za Vrednotenje in Analizo Variant v Večparametrskem Odločanju." Zbornik četrte Elektrotehniške in računalniške konference ERK. Vol. 95, 1995.
17. Žnidaršič, Martin, Marko Bohanec, and Blaž Zupan. "proDEX–A DSS Tool for Environmental Decision-Making." Environmental Modelling & Software 21.10, 2006: 1514-1516.
18. Bihis, Charles. Mastering OAuth 2.0. Packt Publishing Ltd, 2015.
19. Lombardi, Andrew. WebSocket: Lightweight Client-Server Communications. O'Reilly Media, Inc., 2015.
20. Spring Makes Java Simple, Accessed 25.11.2020. https://spring.io/
21. HikariCP. Accessed 25.11.2020. https://github.com/brettwooldridge/HikariCP