



База знаний о состоянии здоровья обучающихся и клиентское приложение для взаимодействия с пользователем

© 2022, С.С. Сосинская✉, И.Д. Сивушков, Д.А. Дубинин, А.С. Дорофеев

Иркутский национальный исследовательский технический университет (ИРНИТУ), Иркутск, Россия

Аннотация

Мониторинг может быть полезным при контроле состояния физического здоровья студента и проводится с целью получения информации, необходимой для принятия управленческих решений по укреплению здоровья обучающихся, занимающихся физической культурой и спортом. В статье описан процесс обработки данных о состоянии физического здоровья обучающихся для использования в автоматизированной экспертной системе на основе онтологии «Здоровье студента», созданной в редакторе *Fluent Editor*. Описаны разработанные онтология и клиентское приложение для интерактивного взаимодействия с пользователем. Цель работы - учёт данных о физическом развитии и подготовленности обучающегося, включая организацию автоматизированной подготовки выходных документов. Для написания клиентского приложения использован язык *Python* и его библиотеки для создания приложения и обеспечения взаимодействия с онтологией. Взаимодействие созданной онтологии с клиентским приложением представляет новизну применяемого подхода.

Ключевые слова: база знаний, онтология, редактор *Fluent Editor*, интерфейс, клиентское приложение.

Цитирование: Сосинская С.С., Сивушков И.Д., Дубинин Д.А., Дорофеев А.С. База знаний о состоянии здоровья обучающихся и клиентское приложение для взаимодействия с пользователем // Онтология проектирования. 2022. Т.12, №3(45). С.325-335. DOI:10.18287/2223-9537-2022-12-3-325-335.

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Введение

По определению экспертов Всемирной организации здравоохранения, «...здоровье – это состояние полного физического, психического и социального благополучия, а не только отсутствие болезней и физических дефектов» [1]. Более развёрнутым является определение, данное в [2]: «Здоровье индивида – это процесс сохранения и развития психических, физиологических, биологических особенностей человека, его оптимальной трудовой способности, социальной активности при максимальной продолжительности активной жизни». Физическое здоровье – это состояние организма, при котором показатели основных физиологических систем лежат в пределах физиологической нормы и адекватно изменяются при взаимодействии человека с внешней средой. В данной статье используется физическое здоровье обучающегося (далее здоровье), которое рассматривается как состояние нормального функционирования всех органов и систем человеческого организма. Физическая подготовленность есть результат физической подготовки к определённому роду деятельности, который характеризуется уровнем развития и владением основного для данной деятельности качества (выносливости, силы, ловкости, гибкости, быстроты).

Наблюдение за состоянием здоровья людей, занимающихся физической культурой и спортом, проводится путём систематических обследований, которые состоят из первичных

до начала занятий физическими упражнениями, повторных - для контроля состояния здоровья и дополнительных обследований перед спортивными соревнованиями, после перенесенных заболеваний или травм, длительных перерывов в занятиях физическими упражнениями. Цель обследований – изучение состояния здоровья для решения вопроса о допуске к занятиям физической культурой, спортом или к участию в соревнованиях. Повторные медицинские обследования позволяют выявлять эффективность воздействия физических упражнений на здоровье и физическое развитие занимающихся этими видами деятельности и вносить соответствующие изменения в занятия.

На основе данных мониторинга физического развития и подготовленности обучающихся ИРНИТУ, накапливаемых в виде *Excel*-таблицы, разработана онтология «Здоровье студента» для хранения экземпляров классов, определённых в онтологии, их изменения и выдачи необходимых выходных документов. Разработано клиентское приложение, рассчитанное на пользователя-непрофессионала, для ввода данных об обучающихся, состоянии их физического здоровья и подготовленности, загрузки данных в онтологию.

1 Показатели физического развития и физической подготовленности обучающихся

Занятия физической культурой или спортом являются эффективным способом реализации физиологических потребностей организма человека, потребности в движении [3].

Одной из основных задач высшего образования является сохранение и укрепление здоровья обучающихся. Для решения данной задачи высшие учебные заведения создают благоприятные условия для всестороннего физического развития студентов. В то же время для многих студентов характерно пассивное отношение к занятиям физической культурой или спортом, хотя их значимость и ценность признаются большинством из них.

Необходимо осуществлять мониторинг физического развития и физической подготовленности, которые определяются следующими показателями.

Показатели физического развития:

- рост, см,
- вес, кг,
- окружность грудной клетки, см,
- жизненная ёмкость легких, мл,
- артериальное давление, мм/рт/ст: систолическое, диастолическое,
- частота сердечных сокращений (ЧСС, раз в 10 сек): до нагрузки, после нагрузки, время восстановления,
- Динамометрия, кг: левой руки, правой руки,
- челночный бег 10 X 5 (сек).

Показатели физической подготовленности:

- проба Штанге (задержка дыхания на вдохе), сек,
- проба Генче (задержка дыхания на выдохе), сек,
- ортостатическая проба: ЧСС лежа, ЧСС стоя, разница в ЧСС
- клиностатическая проба: ЧСС стоя, ЧСС лежа, разница в ЧСС,
- бег 100, 1000, 3000 метров, сек,
- вис/подтягивание, сек,
- подъем туловища за 30 сек, количество раз,
- наклон вперед, см,
- прыжок в длину с места, см,
- пресс в висе за 1 мин, количество раз,
- отжимания за 1 мин, количество раз.

Вести наблюдение и проводить измерения физических показателей здоровья следует регулярно, а полученные данные необходимо рассматривать в динамике. Научно-исследовательская лаборатория мониторинга физического здоровья ИРНИТУ проводит мо-

нитинг физического здоровья обучающихся, ведёт разработку инновационных методов реабилитации здоровья обучающихся, а также занимается совершенствованием учебно-тренировочного процесса подготовки спортсменов [4].

2 Разработка онтологии в редакторе *Fluent Editor*

Для решения названных задач применяется онтологический подход к формированию базы знаний, который является гибким и удобным для моделирования. Онтология рассматриваемой предметной области (ПрО) - её формализованное представление, которое включает словарь терминов ПрО и множество логических связей (типа «элемент-класс», «часть-целое»), которые описывают, как эти термины соотносятся между собой [5-9].

Основными элементами любой онтологии являются:

- классы, описывающие понятия ПрО;
- атрибуты, описывающие свойства классов и их экземпляров;
- экземпляры (объекты), содержащие конкретные значения атрибутов;
- отношения – вид атрибута, определяющий зависимости между классами.

Fluent Editor [10-12] – многофункциональное и интуитивно-понятное приложение, которое позволяет редактировать онтологии, визуализировать связи между понятиями и их экземплярами и выполнять запросы к базе знаний на основе логического вывода. В нём используется язык представления знаний, что позволяет создать точные формализованные описания различных ПрО. Естественно-языковое описание является главным отличием *Fluent Editor* от других онтологических редакторов, что позволяет освоить создание онтологий широкой группе пользователей.

На основе анализа обрабатываемых данных можно выделить следующие классы онтологии «Здоровье студента»: институт, группа, курс, преподаватель, студент, показатели студентов, описанные в разделе 1.

Выходные документы: выборка данных по определённому показателю для заданного курса, группы, преподавателя, студента, института и т.п. Подобные документы позволяют в динамике увидеть процесс изменения показателей обучающихся, группы в целом и т.п., например, за семестр.

Для описания класса используется конструкция вида *Every class is a thing*. Класс находится на вершине дерева иерархии. Имена классов начинаются с малой латинской буквы.

Для описания экземпляров класса используется конструкция вида *Classa is a class*.

В данном предложении «*Classa*» является экземпляром класса «*class*». Имена экземпляров начинаются с заглавной латинской буквы.

Для описания атрибутов используется конструкция вида

Every classa has-name nothing-but (some string value).

В данном предложении атрибутом «*classa*» является «*has-name*», данный атрибут имеет строковый тип. Кроме названий могут быть другие атрибуты.

Для описания отношений между классами используется конструкция вида

Every class has a subclass.

Имена классов, объектов, атрибутов и отношений представлены в таблице 1.

Далее приведены фрагменты онтологии в редакторе *Fluent Editor*.

Предложения, описывающие классы онтологии.

Every instructor is a thing.

Every institute is a thing.

Every group is an institute.

Every student is a group.

Every indicators is a student.

Таблица 1 – Имена классов, объектов, атрибутов и отношений для Про «Здоровье студентов»

Класс	Объекты	Атрибуты	Отношения
<i>instructor</i>	<i>Instructora</i> , <i>Instructorb</i> , ...	<i>has-fio</i> , <i>has-phone-number</i> , <i>has-email</i> , <i>has-position</i>	<i>teaches-to</i> , <i>has</i> , <i>is a university</i>
<i>institute</i>	<i>Insta</i> , <i>Instb</i> , ...	<i>has-name</i> , <i>has-short-name</i>	<i>is a thing</i>
<i>group</i>	<i>Gone</i> , <i>Gtwo</i> , ...	<i>has-name</i> , <i>has-year</i> , <i>has-number</i>	<i>belongs-to</i>
<i>student</i>	<i>Studenta</i> , <i>Studentb</i> , ...	<i>has-fio</i> , <i>has-number-of-the-course-book</i> , <i>has-birth-date</i> , <i>has-medical-group</i>	<i>belongs-to</i> , <i>taught-by</i> , <i>has</i>
<i>indicators</i>	<i>Indka</i> , <i>Indkb</i> , ...	<i>has-semester</i> , <i>has-height</i> , <i>has-weight</i> , <i>has-lungs-vital-capacity</i> , <i>has-chest-volume</i> , <i>has-blood-pressure-systolic</i> , <i>has-blood-pressure-diastolic</i> , <i>has-h-r-before</i> , <i>has-h-r-after</i> , <i>has-recovery-time</i> , <i>has-dynamometry-left</i> , <i>has-dynamometry-right</i> , <i>has-stange-test</i> , <i>has-genche-test</i> , <i>has-orthoprobe-lying</i> , <i>has-orthoprobe-standing</i> , <i>has-orthoprobe-difference</i> , <i>has-clinoprobe-standing</i> , <i>has-clinoprobe-lying</i> , <i>has-clinoprobe-difference</i> , <i>has-shuffle-run-10-x-5-meters</i> , <i>has-run-100-meters</i> , <i>has-bench-press-in-1-minute</i> , <i>has-torso-lift-in-30-seconds</i> , <i>has-forward-bend</i> , <i>has-standing-long-jump</i> , <i>has-run-1000-meters</i> , <i>has-torso-lift-hanging</i> , <i>has-push-up</i> , <i>has-run-3000-meters</i>	<i>belongs-to</i>

Предложения для описания атрибутов.

Атрибуты института.

Every ***institute*** ***has-name*** *nothing-but* (some string value).

Every ***institute*** ***has-short-name*** *nothing-but* (some string value).

Атрибуты группы.

Every ***group*** ***has-name*** *nothing-but* (some string value).

Every ***group*** ***has-year*** *nothing-but* (some string value).

Every ***group*** ***has-number*** *nothing-but* (some string value).

Атрибуты преподавателя.

Every ***instructor*** ***has-fio*** *nothing-but* (some string value).

Every **instructor has-phone-number** nothing-but (some string value).

Every **instructor has-email** nothing-but (some string value).

Every **instructor has-position** nothing-but (some string value).

Атрибуты студента.

Every **student has-fio** nothing-but (some string value).

Every **student has-number-of-the-course-book** nothing-but (some string value).

Every **student has-birth-date** nothing-but (some string value).

Every **student has-medical-group** nothing-but (some string value).

Атрибуты показателей студента.

Every **indicators has-semester** nothing-but (some string value).

Every **indicators has-height** nothing-but (some string value).

Every **indicators has-weight** nothing-but (some string value).

Every **indicators has-lungs-vital-capacity** nothing-but (some string value).

Every **indicators has-chest-volume** nothing-but (some string value).

Every **indicators has-blood-pressure-systolic** nothing-but (some string value).

Every **indicators has-blood-pressure-diastolic** nothing-but (some string value).

Every **indicators has-h-r-before** nothing-but (some string value).

Every **indicators has-h-r-after** nothing-but (some string value).

Every **indicators has-recovery-time** nothing-but (some string value).

Every **indicators has-dynamometry-left** nothing-but (some string value).

Every **indicators has-dynamometry-right** nothing-but (some string value).

Every **indicators has-stange-test** nothing-but (some string value).

Every **indicators has-genche-test** nothing-but (some string value).

Every **indicators has-orthoprobe-lying** nothing-but (some string value).

Every **indicators has-orthoprobe-standing** nothing-but (some string value).

Every **indicators has-orthoprobe-difference** nothing-but (some string value).

Every **indicators has-clinoprobe-standing** nothing-but (some string value).

Every **indicators has-clinoprobe-lying** nothing-but (some string value).

Every **indicators has-clinoprobe-difference** nothing-but (some string value).

Every **indicators has-shuffle-run-10-x-5-meters** nothing-but (some string value).

Every **indicators has-run-100-meters** nothing-but (some string value).

Every **indicators has-bench-press-in-1-minute** nothing-but (some string value).

Every **indicators has-torso-lift-in-30-seconds** nothing-but (some string value).

Every **indicators has-forward-bend** nothing-but (some string value).

Every **indicators has-standing-long-jump** nothing-but (some string value).

Every **indicators has-run-1000-meters** nothing-but (some string value).

Every **indicators has-torso-lift-hanging** nothing-but (some string value).

Every **indicators has-push-up** nothing-but (some string value).

Every **indicators has-run-3000-meters** nothing-but (some string value).

Предложения для описания отношений.

X teaches-to Y if-and-only-if Y taught-by X.

X has Y if-and-only-if Y belongs-to X.

Предложения для добавления объектов и их атрибутов.

Добавление экземпляра и атрибутов института.

Insta is an institute.

Insta has-name equal-to 'Институт информационных технологий и анализа данных'.

Insta has-short-name equal-to 'ИТиАД'.

Предложения для добавления экземпляра и атрибутов преподавателя.

Instructora is an instructor.

Instructora has-name equal-to 'Петров Пётр Петрович'.

Instructora has-phone-number equal-to '666444'.

Instructora has-email equal-to 'petrov@mail.ru'.

Instructora has-position equal-to 'Профессор'.

Предложения для добавления экземпляра и атрибутов группы, а также отношений, связывающих группу с преподавателем и институтом.

Gone is a group.

Gone has-name equal-to 'ЭБМб'.

Gone has-year equal-to '18'.

Gone has-number equal-to '1'.

Gone belongs-to Insta.

Gone taught-by Instructora.

Предложения для добавления экземпляра и атрибутов студента, а также отношения, связывающего студента с группой.

Studenta is a student.

Studenta has-name equal-to 'Сивушков Иван Дмитриевич'.

Studenta has-number-of-the-course-book equal-to '184321'.

Studenta has-birth-date equal-to '27.06.2000'.

Studenta has-medical-group equal-to 'Основная'.

Studenta belongs-to Gone.

Предложения для добавления экземпляра и некоторых атрибутов показателей студента, а также отношения, связывающего показатели со студентом.

Indka is an indicators.

Indka has-height equal-to '182'.

Indka has-weight equal-to '67'.

Indka belongs-to Studenta.

На основе этих предложений с помощью *Fluent Editor* создана иерархия классов, объектов, атрибутов и отношений. Фрагмент дерева иерархии представлен на рисунке 1.

Граф связей, построенный на основе разработанной онтологии, представлен на рисунке 2.

3 Структура клиентского приложения

Для работы с созданной онтологией разработано клиентское приложение, написанное на языке *Python* [13-15] с использованием библиотек *PyQt5*, *CogniPy* и *Pandas* [16-20]. Клиентское приложение позволяет считывать информацию из файла онтологии и при изменениях дописывать новые данные. Приложение содержит модули, предназначенные для добавления и изменения экземпляров, и модуль для формирования выходных документов.

Главное окно приложения представлено на рисунке 3, а структура клиентского приложения - на рисунке 4.

При нажатии кнопки в верхней части главного окна («Институты», «Преподаватели» и т.д.) осуществляется переход на соответствующую

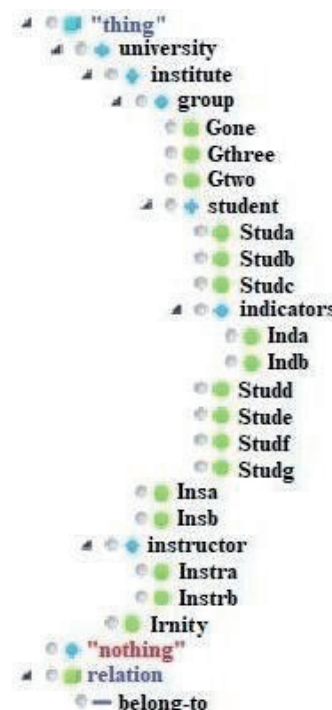


Рисунок 1 – Фрагмент дерева иерархии классов, объектов, атрибутов и отношений

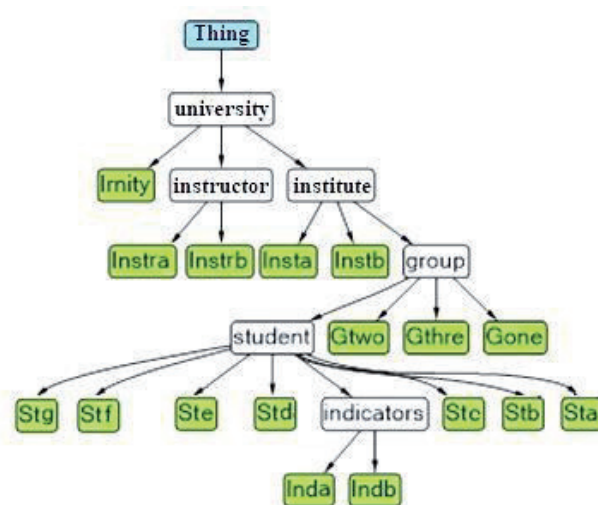


Рисунок 2 – Граф связей для разработанной онтологии

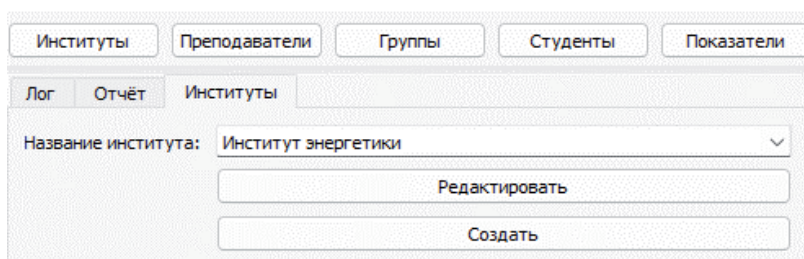


Рисунок 3 – Главное окно приложения

вкладку для работы с определённым классом и его экземплярами. Например, при выборе кнопки «Студенты» отображается окно (рисунок 5). Для добавления или редактирования экземпляра класса, его атрибутов и отношений предназначены соответствующие окна. На рисунке 6 показано окно для добавления данных о студенте.

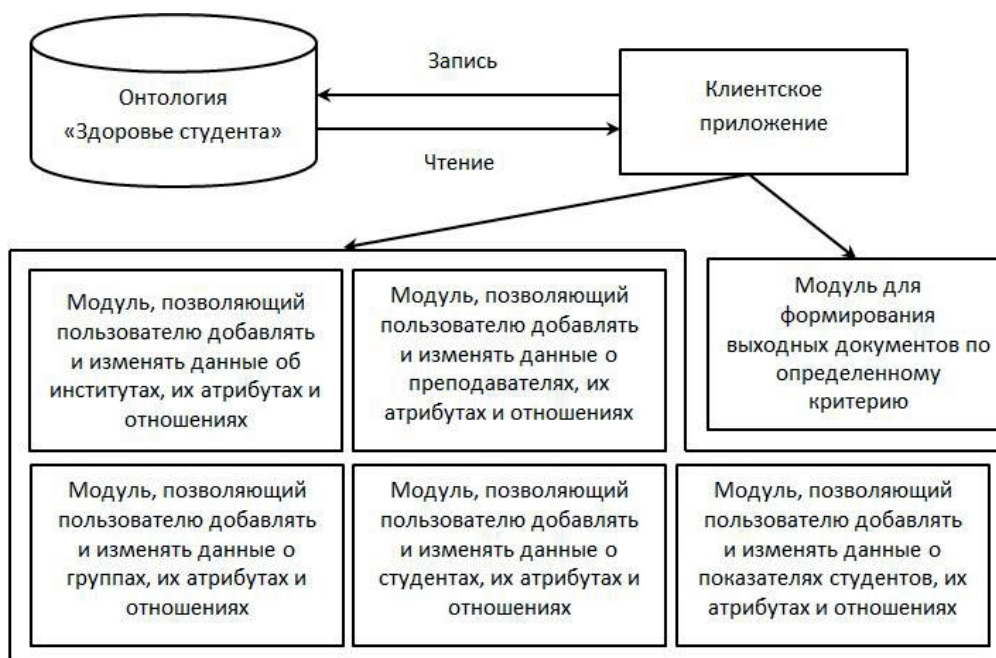


Рисунок 4– Структура клиентского приложения

Рисунок 5 – Окно приложения для выбора экземпляра студента

Рисунок 6 – Окно для добавления данных о студенте с его атрибутами и отношениями

Для идентификации экземпляров каждого класса создаются коллекции имен, начинающихся идентификатором соответствующего класса с добавлением уникального номера.

Модуль приложения, предназначенный для подготовки выходных документов, позволяет получить выборку данных по определённому показателю для заданного курса, группы, преподавателя, института. На рисунке 7 представлена вкладка с результатами выборки по показателю «Бег 100 метров» студентов всех групп и всех курсов, обучающихся у одного преподавателя.

В этом модуле можно получить информацию о всех экземплярах каждого класса (институты, преподаватели, группы, студенты).

Для выбора информации из онтологии используется язык запросов к данным *SPARQL*.

В клиентском приложении определена функция, отправляющая *SPARQL* запрос и возвращающая названия экземпляров определённого класса:

```
def default_query_with_argument(self, argument):
    df = self.onto.sparql_query(
        'SELECT ?entity WHERE'
        '{'
        '?entity rdf:type :'+ argument + ' FILTER NOT EXISTS'
        '{'
        '?entity rdf:type ?c . ?c rdfs:subClassOf+ :'+ argument + ' . FILTER (?c != :'+ argument + ' )'
        '}'
        '}'
    )
    return df
```

ФИО преподавателя	Институт (скор.)	Шифр группы	Год	Номер	ФИО студента	Дата рождения	Медицинская группа	01. Курс	22. Бег 100 м (сек)
Петров Пётр Петрович	ИТИАД	ЭВМ6	18	1	Сивушков Иван Дмитриевич	27.06.2000	Основная	3	15
Петров Пётр Петрович	ИТИАД	ЭВМ6	18	1	Белоусов Денис Андреевич	15.07.2000	Специальная "А"	2	13

Рисунок 7 – Пример выборки данных

Заключение

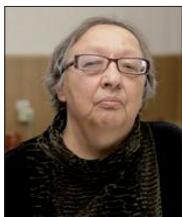
В работе применён онтологический подход к формированию базы знаний, который является гибким и удобным для моделирования. Он позволяет работать с созданной онтологией «Здоровье студента» пользователям, которые не владеют информационными технологиями. Созданное приложение позволяет легко выполнять различные запросы.

Список источников

- [1] Сайт Всемирной организации здравоохранения. <https://www.who.int/ru/about/governance/constitution/>.
- [2] **Казначеев В.П.** Очерки теории и практики экологии человека. М.: Наука. 1983. 260 с.
- [3] **Дюкина Л.А., Игнатъев С.В.** Показатели физического развития и физической подготовки студентов. Сборник научных трудов. Проблемы современного педагогического образования. Вып. 63. Ч.2. Ялта: РИО ГПА, 2019. С.157-160.
- [4] Научно-исследовательская лаборатория мониторинга физического здоровья ИРНТУ. https://www.istu.edu/deyatelnost/nauka/baza/nil/nil_monitoringa_zdorovya/default/.
- [5] **Gruber T.** Towards Principles for the Design of Ontologies Used for Knowledge Sharing. International Workshop on Formal Ontology. Padova, Italy, 1993. P.907-928.
- [6] **Гориков С.** Введение в онтологическое моделирование. ТриниДата, 2016. 165 с. <https://trinidata.ru/files/SemanticIntro.pdf>.
- [7] **Guarino N.** Formal ontology, conceptual analysis and knowledge representation. Int. J. of Human Computer Studies. 1995. Vol. 43(5/6). P.625-640.
- [8] **Lammari N., du Mouza C., Metais E.** POEM: an Ontology Manager based on Existence Constraints. In: S.S. Bhowmick, J. Küng, R. Wagner (eds.): Database and Expert Systems Applications. Proc. 19th Int. Conf. DEXA 2008 (Turin, Italy, September 1-5, 2008). Lecture Notes in Computer Science, vol. 5181. Berlin Heidelberg: Springer-Verlag, 2008. P.81-88.
- [9] **Sertkaya B.** A survey on how description logic ontologies benefit from FCA. In: Proc. of the 7th Int. Conf. on Concept Lattices and Their Applications (Sevilla, Spain, October 19-21, 2010). University of Sevilla, 2010. P.2-21.
- [10] Описание редактора Fluent Editor. <https://www.cognitum.eu/download/download.aspx?id=1001>.
- [11] **Хряцёва О.А., Нестеренко М.В.** Построение онтологической модели рецептов блюд с использованием редактора Fluent Editor / OPEN INNOVATION: сборник статей IX Международной научно-практической конференции. В 2 ч. Ч.1. Пенза: Наука и Просвещение, 2019. С.117-125.
- [12] **Borgest N.M., Orlova A.A.** Ontological modeling of the preparation of flight manuals. *Przegląd Elektrotechniczny*, 2017. Vol.93, No 1. P.105-107.

- [13] Руководство по языку программирования Python. <https://metanit.com/python/tutorial/>
- [14] Язык программирования Python – обучающие материалы. <https://tproger.ru/tag/python/>
- [15] Python 3 Documentation. <https://docs.python.org/3/>
- [16] Qt 5.15 Documentation. <https://doc.qt.io/qt-5.15/>
- [17] PyQt5 Reference Guide. <https://www.riverbankcomputing.com/static/Docs/PyQt5/>
- [18] Pandas Documentation. <https://pandas.pydata.org/docs/>
- [19] CogniPy for Pandas GitHub. <https://github.com/cognitum-octopus/cognipy>
- [20] CogniPy for Pandas Documentation. <https://cognipy.org/index.html>

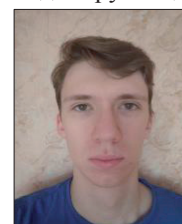
Сведения об авторах



Сосинская Софья Соломоновна, окончила Иркутский государственный университет по специальности «Математик-вычислитель» (1963). К.т.н. (1977), доцент института информационных технологий и анализа данных ИРНИТУ. Область научных интересов – искусственный интеллект, трансляторы, программирование. Имеет свыше 100 публикаций. Под её руководством успешно защитили кандидатские диссертации 4 аспиранта. sosinskaya@mail.ru. ✉.



Сивушков Иван Дмитриевич, окончил бакалавриат ИРНИТУ по направлению «Информатика и вычислительная техника» (2022). Прошёл подготовку в Институте лингвистики и межкультурной коммуникации ИРНИТУ по программе «Переводчик в сфере профессиональной коммуникации». sivushkovivan@gmail.com.



Дубинин Дмитрий Андреевич, окончил ИРНИТУ по специальности «Вычислительные машины, комплексы, системы и сети» (2011). Область научных интересов – искусственный интеллект, автоматизация, экспертные системы. dhdaan.job@gmail.com.



Дорофеев Андрей Сергеевич, окончил Иркутский государственный технический университет (2001), к.т.н. (2008), доцент. Доцент института информационных технологий и анализа данных ИРНИТУ. Автор более 50 публикаций. ORCID: 0000-0002-8498-3301; Author ID (РИНЦ): 189500; Author ID (Scopus): 56349273700. Author ID (Web Of Science): AAM-6919-2021. dorobaik@istu.edu.

Поступила в редакцию 26.07.2022, после рецензирования 26.08.2022. Принята к публикации 04.09.2022.



Knowledge base on the health status of students and a client application for user interaction

© 2022, S.S. Sosinskaya✉, I.D. Sivushkov, D.A. Dubinin, A.S. Dorofeev

Irkutsk National Research Technical University, Irkutsk, Russia

Abstract

Monitoring can be useful in controlling the state of a student's physical health and is carried out in order to obtain the information necessary to make management decisions to improve the health of students involved in physical culture and sports. The article describes the data processing process on the state of physical health of students, used in an automated expert system based on the "Student's Health" ontology and created in Fluent Editor. The developed ontology and client application for interaction with the user are described. The purpose of the work is to take into account data on the physical development and preparedness of the student, including the organization automated preparation of output documents. The Python language and its libraries were used to create an application and ensure interaction with the ontology. The interaction of the created ontology with the client application is a novelty of the applied approach.

Key words: knowledge base, ontology, Fluent editor, interface, graphics application.

For citation: Sosinskaya SS, Savushkov ID, Dubinin DA, Dorofeev AS. Knowledge base on the health status of students and a client application for user interaction [In Russian]. *Ontology of designing*. 2022; 12(1): 325-335. DOI:10.18287/2223-9537-2022-12-3-325-335.

Conflict of interest: The author declares no conflict of interest.

List of figures and table

Figure 1 - Fragment of the hierarchy tree of classes, objects, attributes and relations

Figure 2 - Relations graph for the developed ontology

Figure 3 - Main application window

Figure 4 - Structure of the client application

Figure 5 - Application window for selecting a student

Figure 6 - Window for adding a student with its attributes and relations

Figure 7 - Example of data sampling

Table 1 - Names of Classes, Objects, Attributes and Relationships for the Subject area Student Health

References

- [1] World Health Organization website. <https://www.who.int/ru/about/governance/constitution/>.
- [2] **Kaznacheev VP.** Essays on the theory and practice of human ecology [In Russian]. M.: Science. 1983. 260 p.
- [3] **Dyukina LA, Ignatiev SV.** Indicators of physical development and physical training of students. Problems of modern pedagogical education [In Russian]. Collection of scientific papers. Issue. 63. Part 2. Yalta: RIO GPA, 2019. P.157-160.
- [4] Official website of IRNITU. Research Laboratory of Physical Health Monitoring. https://www.istu.edu/deyatelnost/nauka/baza/nil/nil_monitoringa_zdorovya/default/.
- [5] **Gruber T.** Towards Principles for the Design of Ontologies Used for Knowledge Sharing. International Workshop on Formal Ontology. Padova, Italy, 1993. P.907-928.
- [6] **Gorshkov S.** Introduction to ontological modeling [In Russian]. TriniData, 2016. 165 p. <https://trinidata.ru/files/SemanticIntro.pdf>.
- [7] **Guarino N.** Formal ontology, conceptual analysis and knowledge representation. *Int. J. of Human Computer Studies*. 1995; 43(5/6): 625-640.
- [8] **Lammari N, du Mouza C, Metais E.** POEM: an Ontology Manager based on Existence Constraints. In: S.S. Bhowmick, J. Küng, R. Wagner (eds.): Database and Expert Systems Applications. Proc. 19th Int. Conf. DEXA

- 2008 (Turin, Italy, September 1-5, 2008). Lecture Notes in Computer Science, vol. 5181. Berlin Heidelberg: Springer-Verlag, 2008. P.81-88.
- [9] **Sertkaya B.** A survey on how description logic ontologies benefit from FCA. In: Proc. of the 7th Int. Conf. on Concept Lattices and Their Applications (Sevilla, Spain, October 19-21). University of Sevilla, 2010. P.2-21.
- [10] Description of Fluent editor. <https://www.cognitum.eu/download/download.aspx?id=1001>.
- [11] **Khryashcheva O.A., Nesterenko M.V.** Building an ontological model of food recipes using the Fluent Editor / OPEN INNOVATION: a collection of articles from the IX International Scientific and Practical Conference. At 2 p.m. Part 1. Penza: Science and Education, 2019. P.117-125.
- [12] **Borgest NM, Orlova AA.** Ontological modeling of the preparation of flight manuals. *Przegląd Elektrotechniczny*, 2017; 93(1): 105-107.
- [13] Guide to the Python programming language. <https://metanit.com/python/tutorial/>
- [14] The Python Programming Language - Tutorials. <https://tproger.ru/tag/python/>
- [15] Python 3 Documentation. <https://docs.python.org/3/>
- [16] Qt 5.15 Documentation. <https://doc.qt.io/qt-5.15/>
- [17] PyQt5 Reference Guide. <https://www.riverbankcomputing.com/static/Docs/PyQt5/>
- [18] Pandas Documentation. <https://pandas.pydata.org/docs/>
- [19] CogniPy for Pandas GitHub. <https://github.com/cognitum-octopus/cognipy>
- [20] CogniPy for Pandas Documentation. <https://cognipy.org/index.html>.
-

About the authors

Sofya Solomonovna Sosinskaya, graduated from Irkutsk State University with a degree in Mathematician-Computer (1967), PhD (1977), Associate Professor at the Institute of Information Technologies and Data Analysis. Research interests are artificial intelligence, translators, and programming. She has over 100 publications. Four PhD students successfully defended their dissertations under her guidance. sosinskaya@mail.ru. ✉.

Ivan Dmitrievich Sivushkov, graduated from the Irkutsk National Research Technical University with bachelor's degree in "Informatics and Computer Engineering" (2022). Passed professional training at the Institute of Linguistics and Intercultural Communication of INRTU under the program "Translator in the field of professional communication". sivushkovivan@gmail.com.

Dmitry Andreevich Dubinin, graduated from the Irkutsk State Technical University with a degree in Computers, Complexes, Systems and Networks. Research interests are artificial intelligence, automation, and expert systems. dhdaan.job@gmail.com.

Andrey Sergeevich Dorofeev, graduated from the Irkutsk State Technical University (2001), PhD (2007), Associate Professor. He is currently an Associate Professor at the Institute of Information Technology and Data Analysis. Author of more than 50 publications. ORCID: 0000-0002-8498-3301; Author ID (RSCI): 189500; Author ID (Scopus): 56349273700. Author ID (Web Of Science): AAM-6919-2021. dorbaik@istu.edu.

Received July 26, 2022. Revised August 26, 2022. Accepted September 04, 2022.
