

УДК 004.414.3

ГЕНЕРАЦИЯ ИНДИВИДУАЛЬНЫХ ОБРАЗОВАТЕЛЬНЫХ ТРАЕКТОРИЙ И РАСПИСАНИЯ ОБУЧЕНИЯ В ПАРАДИГМЕ ИНДИВИДУАЛИЗАЦИИ ОБРАЗОВАНИЯ

М. М. Абрамский¹, Э. Ф. Батырова², А. Р. Марданова³, Т. А. Ахметзянова⁴

¹⁻⁴ *Высшая школа информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета*

¹ma@it.kfu.ru, ²elvira3805@gmail.com, ³aigulmardanova96@gmail.com,

⁴tanya.push0494@gmail.com

Аннотация

Представлен подход к индивидуализации образования, основанный на автоматизированной генерации индивидуальной образовательной траектории и расписания, учитывающих особенности каждого обучающегося и его пожелания. Описан принцип действия разработанных инструментов генерации. Затронуты вопросы применения разработанных подходов и инструментов в высшем образовании.

Ключевые слова: *smart-образование, индивидуализация образования, индивидуальный учебный план, индивидуальная траектория, генетические алгоритмы, генерация расписания*

ВВЕДЕНИЕ

Согласно федеральной целевой программе развития образования одной из важных задач является «индивидуализация образовательных траекторий с учетом личностных свойств, интересов и потребностей обучающегося» [1].

Под *индивидуализацией* понимается организация учебного процесса, учитывающая индивидуальные особенности каждого студента. Это помогает создать оптимальную среду для максимальной реализации возможностей обучающегося [2].

Под *индивидуальной образовательной траекторией* в широком смысле понимается спроектированный на основе компетенций и характеристик обуча-

ющего перечня дисциплин и проектов, который студент должен пройти для достижения наилучших для себя образовательных результатов.

В рамках решения задач составления индивидуальных траекторий все большее развитие получают концепция *Smart University* («*Smart-университет*», «*Умный университет*») и ее неотъемлемая часть, посвященная непосредственно образованию, – *Smart Education* (*Smart-образование*), предполагающая адаптивную реализацию образовательного процесса с применением интеллектуальных (*smart*) цифровых инструментов, осуществляющих индивидуализацию образовательной траектории с помощью средств машинного обучения, анализа данных, визуализации и др.

Реализация концепции *Smart University*, в том числе и *Smart Education*, является одной из стратегических инициатив Казанского (Приволжского) федерального университета (КФУ), что отражено в 4-м этапе Плана мероприятий по реализации программы повышения конкурентоспособности КФУ [3].

Концепция *Smart-университета*, описывающая, в том числе, вопросы кампуса, электронного образования, коллаборации университетов для объединения ресурсов, подробно описана в [4].

Концепция *Smart Education* в литературе понимается достаточно широко. Например, в работе [5] к *Smart-образованию* относят, в том числе, вопросы организации смешанной формы обучения, предполагающей синергию обучения в классе и *online-обучения*, вопросы организации исследовательской, познавательной и проектной деятельности студентов, особенности механизмов поддержания актуальных компетенций студентов и преподавателей, соответствующих критериям рынка труда. В работе [6] достаточно подробно изучены вопросы технической реализации облачной системы управления образовательным контентом.

Не умаляя всех достоинств концепции *Smart Education*, в данной работе будем рассматривать именно вопросы индивидуализации и ее воплощения в образовательный процесс вуза с помощью специальных цифровых инструментов, при этом будут затронуты и смежные компоненты концепции.

Работа построена следующим образом. В первом разделе описан предлагаемый авторами подход к реализации индивидуализации в вузе. Во втором разделе разобраны вопросы генерации индивидуальных образовательных тра-

екторий, третий раздел касается механизмов сбора сведений, необходимых для генерации расписания обучения, а также алгоритма генерации.

1. ПРЕДЛАГАЕМЫЙ ПОДХОД К ИНДИВИДУАЛИЗАЦИИ ОБРАЗОВАНИЯ

В данной работе под *индивидуальным учебным планом* и *индивидуальной образовательной траекторией* будет подразумеваться одно и то же понятие. Отметим все же, что под индивидуальной образовательной траекторией понимается больше содержательная часть индивидуального образования, а под индивидуальным учебным планом – ее реализация в качестве формального документа. Однако в рамках данной работы это различие не является существенным.

Учебный план и расписание занятий являются неотъемлемой частью образовательного процесса. В рамках индивидуализации компетенции учебного плана, предусмотренные профильными ФГОСами, должны строго коррелировать с индивидуальной образовательной траекторией каждого студента. Вместе с этим эти же компетенции должны соответствовать требованиям рынка труда, что отражено во внедрении в высшее образование профессиональных стандартов (ФЗ №122 от 2 мая 2015 г. об обязательном использовании профессиональных стандартов при установлении требований к профессиональным компетенциям выпускников) [7].

Имеет место постоянная работа по адаптации учебного плана к актуальным требованиям рынка, иногда выходящим за пределы профессиональных стандартов. При внедрении индивидуализации происходит переход от общего учебного плана к индивидуальному для каждого студента, и работа по адаптации всех индивидуальных учебных планов, очевидно, является трудозатратной.

При генерации расписания на основе составленного индивидуального учебного плана необходимо учитывать дополнительную информацию: личные предпочтения студентов и преподавателей, совместимость их компетенций и интересов, характеристики аудиторного фонда. Данная задача также не может быть решена без разработки специальных информационных систем и цифровых инструментов, которые смогут решить ее автоматически.

Принципиальная схема работы разрабатываемых инструментов представлена на рисунках 1 и 2.

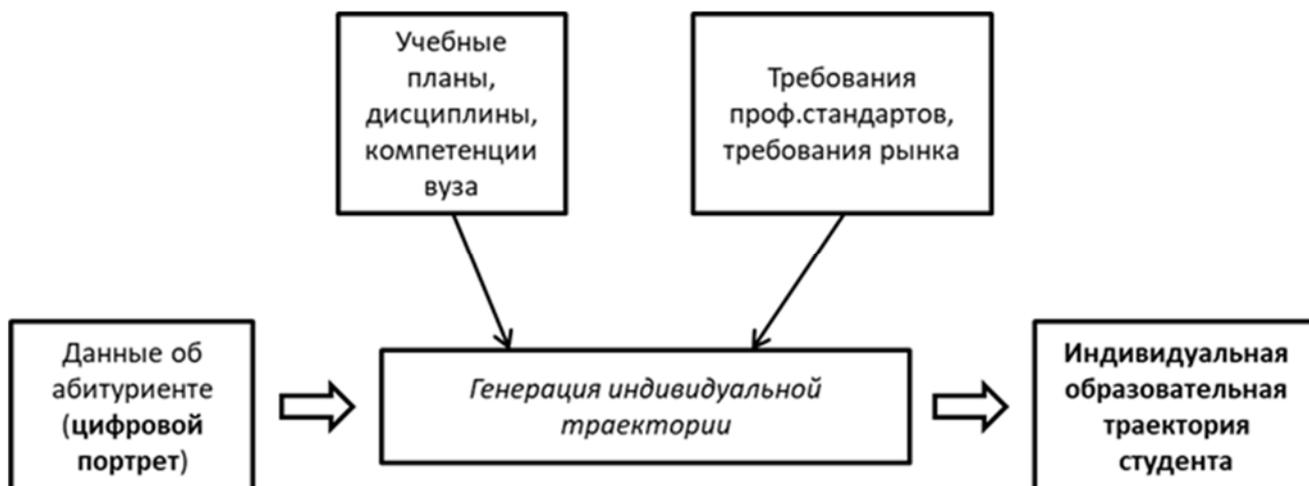


Рис. 1. Принцип работы модуля генерации индивидуальной образовательной траектории



Рис. 2. Принцип работы платформы сбора требований к расписанию и его генерации

Отметим, что вопросы устройства цифрового портрета обучающегося требуют отдельного рассмотрения, что будет сделано в последующих публикациях. На данный момент цифровым портретом студента можно считать хранимую в базе данных информацию, включающую перечень его оценок аттестата средней школы, баллы ЕГЭ, иные достижения и умения студента. Все данные портрета обучающегося хранятся в формате «строка/значение» – «компетенция/уровень владения», «предмет/оценка» или «умение/уровень».

2. РЕАЛИЗАЦИЯ ГЕНЕРАЦИИ ИНДИВИДУАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ТРАЕКТОРИИ ОБУЧАЮЩЕГОСЯ

Для построения индивидуальных учебных планов разработан алгоритм, который изображен на рис. 3.

Алгоритм начинается с обработки цифрового портрета абитуриента, в данном случае – перечня знаний, умений, навыков, компетенций и достижений. Также на начальном этапе алгоритма происходит извлечение компетенций из ФГОС и учебных планов, а также трудовых функций из профессиональных стандартов. В будущем планируется добавить в этот этап автоматизированную обработку вакансий и трендов рынка труда.

Следующим шагом производится вычисление корреляции элементов цифрового портрета с компетенциями учебного плана с целью обнаружения схожести этих значений. Сравнение производится с помощью алгоритма шинглов [8] – поиска копий и дубликата. Предварительно каждая строка очищается от знаков препинания и стоп-слов (предлогов, союзов, местоимений). Затем каждая строка представляется в виде множества n -грамм – последовательностей слов фиксированной длины n (или шинглов). С целью увеличения эффективности шинглы из текстовых подстрок преобразуются в числа – дактилограммы. Элементы двух множеств сравниваются по количеству схожих элементов. В алгоритме установлен лимит количества схожих элементов, при превышении которого делается вывод о соответствии навыка студента и компетенции в учебном плане.

Здесь сразу же стоит отметить, что в настоящее время абитуриент самостоятельно выбирает, по какой образовательной программе он хочет учиться, и, если его результаты единого государственного экзамена (ЕГЭ) соответствуют проходному баллу, то он автоматически поступает на выбранное направление. Соответствие между уровнем компетенций студента и возможностью учиться по выбранному направлению определяется лишь перечнем из 3 или 4 результатов ЕГЭ по общеобразовательным предметам, требуемым для данного направления.

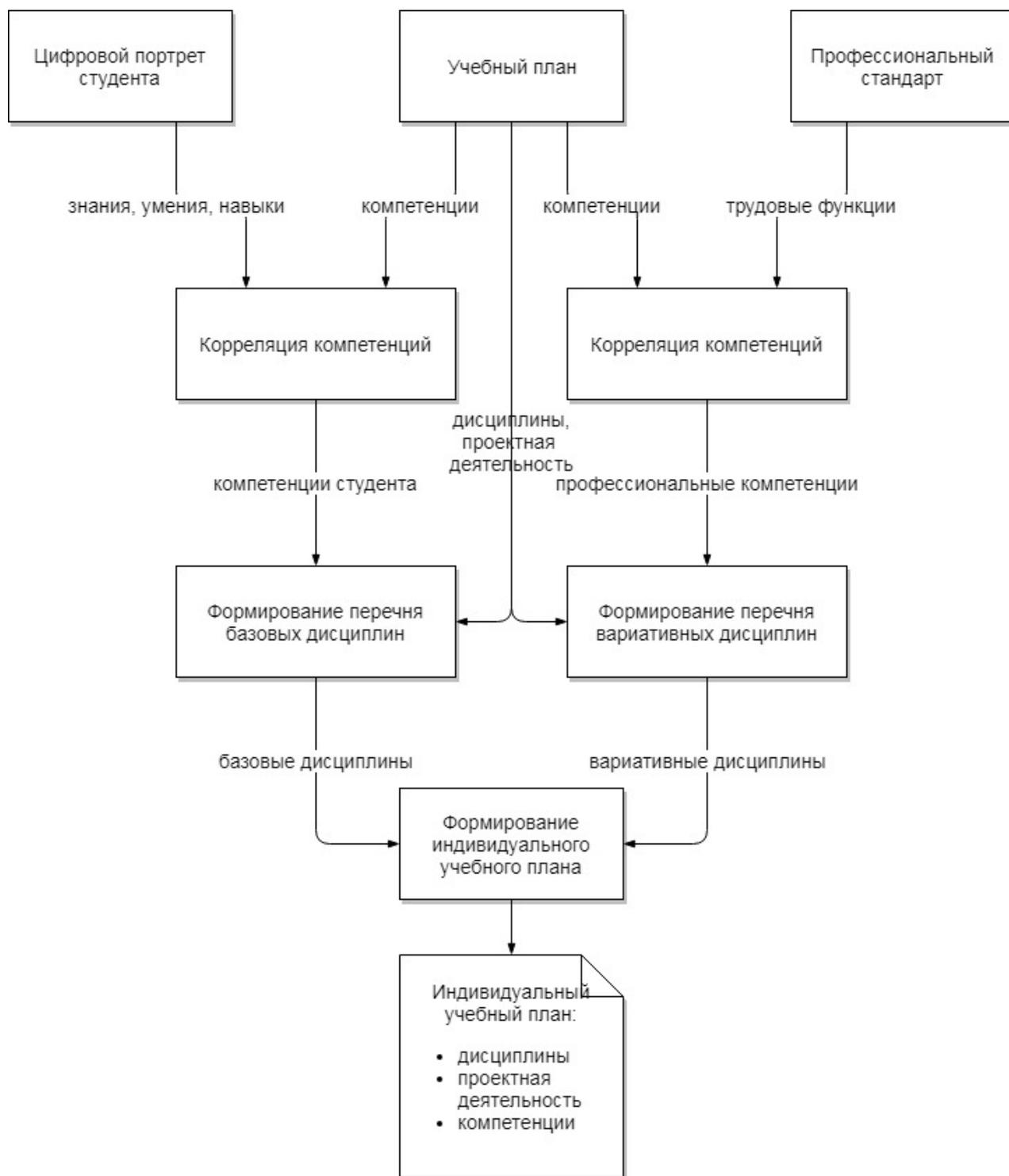


Рис. 3. Алгоритм построения индивидуальных учебных планов

В настоящее время стало очевидным, что оценки за ЕГЭ не являются исчерпывающим описанием компетенций абитуриента, особенно в рамках индивидуализации образования. В поиске более тесных связей между компетенция-

ми студента (чем он владеет) и компетенциями учебного плана (чем он может овладеть) и заключается описанный этап вычисления корреляции.

Параллельно теми же средствами происходит аналогичное вычисление корреляции трудовых функций из профессиональных стандартов с компетенциями учебных планов. Трудовые функции определяются профессиональной деятельностью, которой студент желает заниматься после окончания университета. Таким образом, для трудовых функций подбираются профессиональные компетенции из учебных планов.

На выходе первого шага алгоритма имеется перечень компетенций, которые обучающийся (на основе своих текущих навыков) сможет сформировать, обучаясь в вузе. Отметим, что компетенции не обязательно должны соответствовать только одному учебному плану. Наоборот, корреляции с несколькими учебными планами позволят создать индивидуальный план, способствующий подготовке востребованного междисциплинарного специалиста.

Среди подходящих учебных планов выбирается учебный план, обозначаемый как базовый. Наличие базового учебного плана необходимо, поскольку по формальным правилам индивидуальный учебный план должен быть встроен в один из базовых учебных планов. Подразумевается, что из базового учебного плана в индивидуальный учебный план попадут базовые предметы, такие, как, например, философия, информатика, физика.

По профессиональным компетенциям подбираются дисциплины, которые могут их сформировать. Данные дисциплины называются вариативными. Отметим, что в рамках настоящей работы к вариативным дисциплинам мы относим также дисциплины по выбору. В будущих работах планируется возможность автоматического составления формата проектной деятельности для развития профессиональных компетенций.

Таким образом, по завершении работы алгоритма имеется перечень дисциплин, формирующих компетенции, учитывающие как потребности студента, так и требования профессионального стандарта. Вместе они представляют собой индивидуальный учебный план.

Модуль с предложенным выше алгоритмом разработан на языке программирования Java с использованием фреймворка Spring [9]. Для хранения

данных была использована СУБД PostgreSQL [10]. Для анализа русскоязычного текста была использована утилита MyStem [11].

3. СБОР СВЕДЕНИЙ ДЛЯ ГЕНЕРАЦИИ РАСПИСАНИЯ И АЛГОРИТМ ГЕНЕРАЦИИ

В качестве связующего звена между составленным индивидуальным планом и генерацией расписания была разработана специальная платформа, обеспечивающая алгоритм генерации расписания необходимыми данными – как путем ручного сбора пожеланий к расписанию со стороны студентов и преподавателей, так и с помощью автоматического создания пожеланий и требований к расписанию на основе анализа цифровых портретов студентов и преподавателей, индивидуальных траекторий, формальных ограничений, соображений эффективности и др. Пример страницы ввода пожелания представлен на рис. 4.

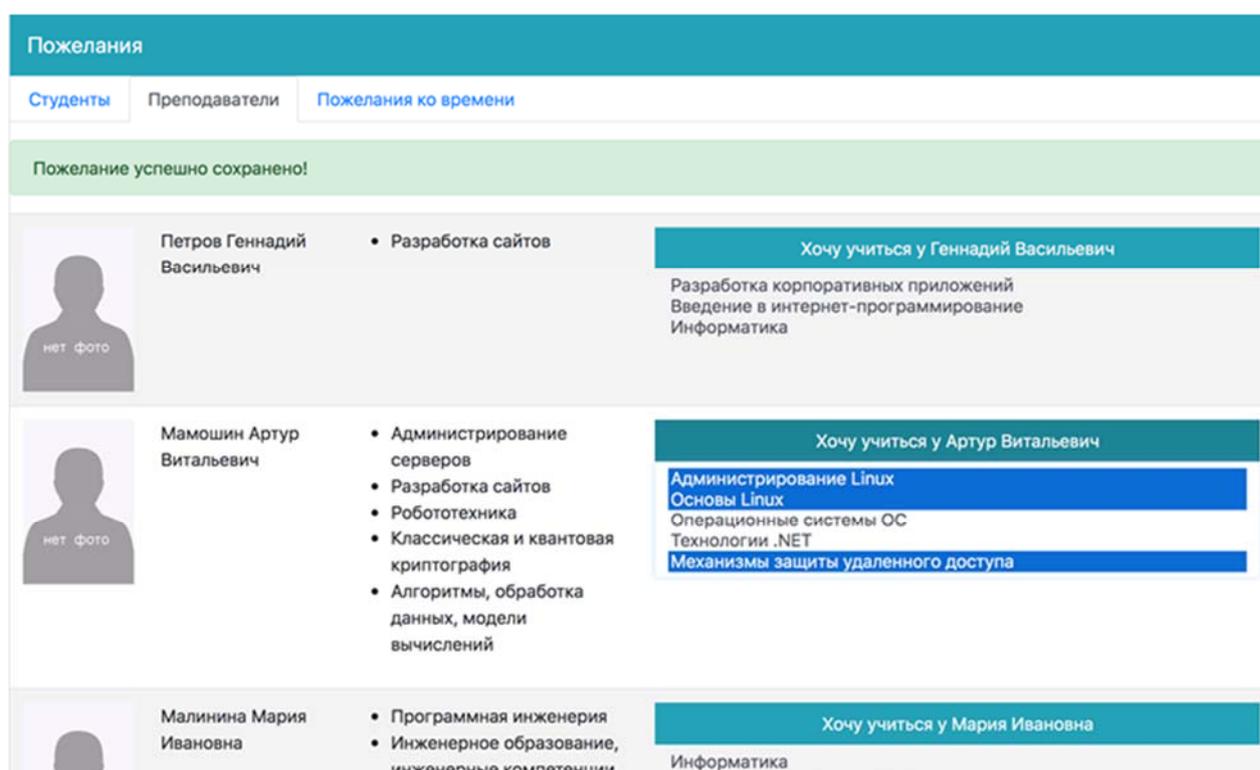


Рис. 4. Страница ввода пожеланий к обучению у конкретного преподавателя по конкретному предмету

Пожелание представляет собой условие к расписанию, выполнение которого либо обязательно, либо желательно для целей индивидуализации, и кото-

рое опирается на составные части цифровых портретов сущностей университета. Был составлен расширенный (в контексте индивидуализации) каталог возможных пожеланий и требований (рис. 5), звездочкой (*) отмечены пожелания и требования, для которых доступна автоматическая генерация.

№	Тип	Название
1	Студент – Студент (*)	Пожелание студента к студенту
2	Студент – Студент – Предмет (*)	Пожелание студента к студенту для определенного предмета
3	Студент – Преподаватель	Пожелание студента к преподавателю
4	Студент – Преподаватель – Предмет	Пожелание студента к преподавателю для определенного предмета
5	Преподаватель – Студент (*)	Пожелание преподавателя к студенту
6	Преподаватель – Студент – Предмет (*)	Пожелание преподавателя к студенту для определенного предмета
7	Пользователь – Начало занятий	Пожелание ко времени первой пары (не раньше, чем)
8	Пользователь – Начало занятий – Предмет	Пожелание ко времени первой пары (не раньше, чем) для определенного предмета
9	Пользователь – Конец занятий	Пожелание ко времени последней пары (не позже, чем)
10	Пользователь – Конец занятий – Предмет	Пожелание ко времени последней пары (не позже, чем) для определенного предмета
11	Преподаватель – Аудитория (*)	Пожелание преподавателя к аудитории для определенного предмета
12	Преподаватель – Оборудование	Пожелание преподавателя к наличию оборудования в аудитории для определенного предмета
13	Предмет – Оборудование	Требования к наличию оборудования в аудитории для определенного предмета
14	Предмет – Аудитория (*)	Пожелание к аудитории для определенного предмета

Рис. 5. Каталог пожеланий и требований. (*) – доступна автоматическая генерация

Приведем пример автоматической генерации пожелания/требования «Студент–студент». Предположим, что в рамках генерации необходимо собрать в группы студентов, которые имеют в своем цифровом портрете и индивидуальных траекториях схожие компетенции. Для оценки «схожести» была предложена достаточно простая формула:

$$similarityC = |PC1 \cap C2| + |C1 \cap PC2| + |C1 \cap C2|,$$

где PC1 – множество личных компетенций первого студента (из цифрового портрета), PC2 – множество личных компетенций второго студента, C1 – множество развиваемых компетенций первого студента (из индивидуального учебного плана), C2 – множество развиваемых компетенций второго студента.

При включении параметра «автоматическая генерация» для данного требования в алгоритм генерации расписания попадут условия сочетания студентов по величине данного признака. Отметим, что данная формула легко поддается изменению на другую в рамках системы.

Оговоримся, что целью данной работы не является выработка критериев наиболее эффективного распределения студентов. К примеру, возможно будет

эффективным распределить иностранных студентов, владеющих английским языком, к преподавателю, который тоже им владеет. Или же наоборот, возможно, для развития их компетенций коммуникации на русском языке эффективнее будет распределить их к преподавателю, не владеющему английским языком.

Разрабатываемые инструменты позволяют делать гибкую настройку этих критериев, а их наполнение с целью сделать распределение эффективным с той или иной точки зрения отдается полностью пользователям-экспертам, управляющим алгоритмом генерации расписания.

Задача составления расписания неоднократно решалась для классических случаев, на рынке существует целая серия приложений, генерирующих расписание занятий [12–14]. Также осуществлялись попытки ее решения с учетом некоторых особенностей, например, пожеланий преподавателей [15]. Недостатком существующих решений является их исходное несоответствие принципу индивидуализации образования, а также разработанной выше модели.

В качестве способа реализации алгоритма генерации расписания в контексте индивидуализации нами был выбран генетический алгоритм, зарекомендовавший себя как надежный подход к решению задачи составления расписания [16]. Этот способ позволяет строить сложные конструкции путем генерации начальной популяции и скрещивания между элементами с целью последующего улучшения.

Схема разработанного алгоритма представлена на рис. 6. Первое поколение расписаний строится на основе сущностей университета, конфигурации расписания и системных ограничений.

Конфигурация расписания содержит такие параметры, как количество учебных дней в неделе, количество пар в день, максимальное число студентов в группе. Системные ограничения – это такие ограничения, без которых расписание не может быть построено. Существует три типа системных ограничений: преподаватель не может одновременно находиться на двух парах, студент не может одновременно находиться на двух парах, аудитория не может быть занята одновременно двумя парами. Системные ограничения также проверяются после каждой итерации алгоритма, за счет чего расписание всегда остается корректным.

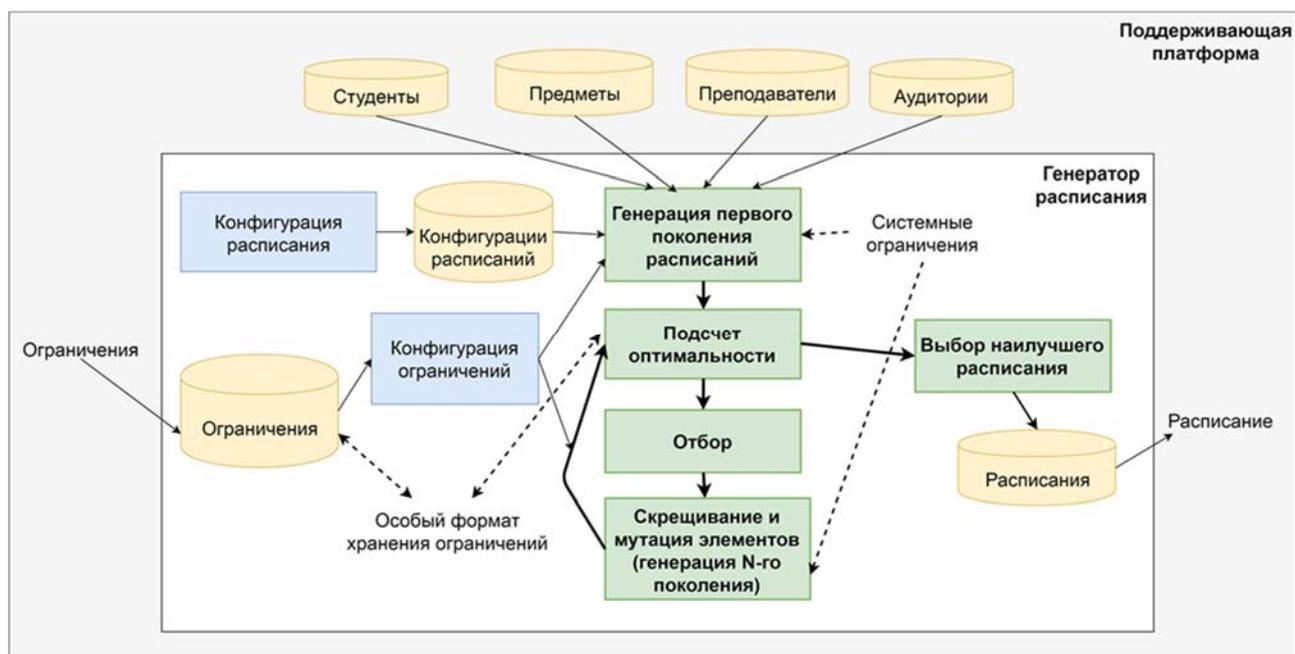


Рис. 6. Схема алгоритма генерации расписания. Под ограничениями понимаются все требования к расписанию, в том числе и пожелания

Подсчет оптимальности расписаний происходит с помощью таблицы, строки которой содержат участвующие в ограничении сущности и boolean-колонку, отображающую выполнение или невыполнение условия.

Тип	Включен	Всего	Выполнено
Студент - Студент - Предмет	<input checked="" type="checkbox"/>	20	7
Студент - Преподаватель	<input checked="" type="checkbox"/>	20	5
Преподаватель - Начало пар - Конец пар	<input checked="" type="checkbox"/>	20	8

Рис. 7. Возможность управления ограничениями в процессе генерации расписания

В ходе работы на алгоритм можно влиять путем включения и выключения типов ограничений (рис. 7). Так можно сначала включить только важные типы ограничений, а когда они будут выполнены, добавить остальные. На данный момент реализованы выключения/включения ограничений по их типу, в будущем планируется проработать вопрос управления отдельными ограничениями.

По окончании работы генератор выбирает расписание с наибольшим процентом выполненных условий. Полученное расписание сохраняется в базу данных, откуда его можно выгрузить для просмотра (рис. 8).

Аудитория	Время	Предмет	Преподаватель	Студенты
1301	2	Информатика	Абрамский Михаил Михайлович	Усачев Артем\ Забиров Салават\ Евдокименко Дмитрий\ Сафина Алия\ Батырова Эльвира Феликсовна\ Тлитова Алина\
1301	3	Математический анализ	Широкова Елена Александровна	Закжовой Илья\ Матросов Кирилл\ Хисамов Данил\ Батырова Эльвира Феликсовна\ Зиганшин Руслан\ Москеева Алина\ Усачев Артем\

Рис. 8. Просмотр расписания. В колонке «время» указан номер пары по расписанию учебного заведения (для КФУ: 2 – 10:10, 3 – 11:50)

Было проведено тестирование модуля генерации расписания на шестидневной учебной неделе с шестью парами в день. Общее количество студентов в тесте – 144 (студентов в группе – максимум 25), количество преподавателей – 15, предметов – 15. Для сравнения каждый тип ограничения был протестирован отдельно (по 20 ограничений в каждом тесте). Наименьший процент выполнения получали пожелания от студентов, а наивысший – от преподавателей. Это объясняется тем, что учесть пожелания каждого студента из группы сложнее, чем пожелание одного преподавателя. Также был произведен общий тест из 140 ограничений различных типов, 80% из которых были выполнены. Это значит, что алгоритм способен учитывать значительную долю пожеланий и рекомендаций.

Платформа для сбора пожеланий и генерации требований, а также модуль генератора расписания были реализованы в виде веб-приложения. Клиентская часть написана на фреймворке Angular [17], что позволяет динамически следить за процессом генерации, а серверная – с использованием Spring Framework. Для хранения данных также использовалась СУБД PostgreSQL.

ЗАКЛЮЧЕНИЕ

Реализованы модуль для построения индивидуальных учебных планов, платформа для сбора пожеланий и требований к расписанию, а также модуль генерации расписания. Представлена модель индивидуализации образования, которую разработанные инструменты могут обеспечить. Проведены эксперименты по проверке работоспособности инструментов на тестовых данных.

В дальнейшем планируется развитие разработанных инструментов в сторону гибкости, удобства использования и большей функциональности. Будет поднят вопрос об эффективной визуализации индивидуальной образовательной траектории и расписания обучения. Будут подробнее рассмотрены цифровой портрет студента и механизмы его формирования.

Работа выполнена в рамках деятельности лаборатории Smart Education Lab Высшей школы информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета.

СПИСОК ЛИТЕРАТУРЫ

1. Концепция Федеральной целевой программы развития образования на 2016–2020 годы / Распоряжение Правительства Российской Федерации от 29.12.2014 № 2765-р. URL: <http://government.ru/media/files/mlorxfXbbCk.pdf>
2. *Cheng, Yin Cheong*. New Paradigm for Re-engineering Education: Globalization, Localization and Individualization. Springer, 2005. 26 p.
3. План мероприятий по реализации программы повышения конкурентоспособности ФГАОУ ВО «Казанский (Приволжский) федеральный университет» на 2013–2020 годы (4-й этап – 2018–2020 гг.). URL: http://kpfu.ru/portal/docs/F_1176208685/DK.KFU_4.etap._utverzh.pdf
4. Smart Universities. Concepts, systems and Technologies. V.L. Uskov, J.P. Bakker, R.J. Howlet, L.C. Jain (Eds.). Springer Inter. Publishing, 2018. 435 p.
5. *Тихомиров В.П., Днепровская Н.В.* Смарт-образование как основная парадигма развития информационного общества // Современные информационные технологии и ИТ-образование. 2015. Т. 1. № 11. С. 9–13.
6. *Ji-Seong Jeong, Mihye Kim and Kwan-Hee Yoo*. A Content Oriented Smart Education System based on Cloud Computing // Int. J. of Multimedia and Ubiquitous

Engineering. 2013. Vol. 8, No. 6. P. 313–328. URL: <http://dx.doi.org/10.14257/ijmue.2013.8.6.31>

7. Федеральный закон от 02.05.2015 г. № 122-ФЗ «О внесении изменений в Трудовой кодекс Российской Федерации и статьи 11 и 73 Федерального закона «Об образовании в Российской Федерации». URL: <http://static.kremlin.ru/media/acts/files/0001201505020007.pdf>

8. Broder A. On the resemblance and containment of documents // Compression and Complexity of Sequences (SEQUENCES'97). IEEE Computer Society, 1998. P. 21–29.

9. Spring Framework. URL: <https://projects.spring.io/spring-framework>

10. PostgreSQL: The World's Most Advanced Open Source Relational Database. URL: <https://www.postgresql.org/>

11. MyStem. Yandex URL: <https://tech.yandex.ru/mystem/>

12. Bullet Education Scheduling and Timetabling. URL: <https://www.bulletsolutions.com/education-scheduling-timetabling/>

13. Mimosa Scheduling Software. URL: <http://www.mimosasoftware.com/>

14. Snap Schedule. URL: <https://www.snapschedule.com/Industry/Education/>

15. «Официальный сайт программного продукта БИТ.ВУЗ.РАСПИСАНИЕ». URL: <http://www.pulsar.ru/progs/1904/>

16. Lim C., Sim E. Production Planning in Manufacturing / Remanufacturing Environment using Genetic Algorithm // GECCO 2005: Proc. of the 2005 Conference on Genetic and Evolutionary Computation, 2005. P. 2217–2218. URL: <http://www.cs.bham.ac.uk/~wbl/biblio/gecco2005/docs/p2217.pdf>

17. Официальный сайт фреймворка Angular 5. URL: <https://angular.io/>

GENERATION OF INDIVIDUAL EDUCATIONAL ROUTES AND LEARNING SCHEDULE IN INDIVIDUALIZATION PARADYGM

M. M. Abramskiy¹, E. F. Batyrova², A. R. Mardanova³, T. A. Akhmetzyanova⁴

¹⁻⁴ Higher School for Information Technologies and Intelligent Systems of Kazan (Volga Region) Federal University

¹ma@it.kfu.ru, ²elviraa3805@gmail.com, ³aigulmardanova96@gmail.com,

⁴tanya.push0494@gmail.com

Abstract

An approach for individualization of education based on automatic generation of individual educational routes and schedule with respect to individual features and wishes of student. A working principle of developed instruments is shown. Question of the application of developed approaches and instruments in higher education are raised.

Keywords: *Smart Education, individualization of education, individual educational program, individual educational route, genetic algorithms, schedule generation*

REFERENCES

1. Kontsepsiya Federal'noj tselevoj programmy razvitiya obrazovaniya na 2016-2020 gody / Rasporyazhenie Pravitel'stva Rossijskoj Federatsii ot 29.12.2014 No 2765-r. URL: <http://government.ru/media/files/mlorxfXbbCk.pdf>
2. *Cheng Yin Cheong*. New Paradigm for Re-engineering Education: Globalization, Localization and Individualization. Springer, 2005. 26 p.
3. Plan meropriyatij po realizatsii programmy povysheniya konkurentosposobnosti FGAOU VO «Kazanskij (Privolzhskij) federal'nyj universitet» na 2013–2020 gody (4-j etap – 2018–2020 gg.). URL: http://kpfu.ru/portal/docs/F_1176208685/DK.KFU_4.etap._utverzh.pdf
4. Smart Universities. Concepts, systems and Technologies. V.L. Uskov, J.P. Bakker, R.J. Howlet, L.C. Jain (Eds.). Springer Inter. Publishing. 2018. 435 p.
5. *Tikhomirov V.P., Dneprovskaya N.V.* Smart-obrazovanie kak osnovnaya paradigma razvitiya informatsionnogo obshhestva // *Sovremennye informatsionnye tekhnologii i IT-obrazovanie*. 2015. T. 1. № 11. S. 9–13.

6. *Ji-Seong Jeong, Mihye Kim and Kwan-Hee Yoo.* A Content Oriented Smart Education System based on Cloud Computing // *Int. J. of Multimedia and Ubiquitous Engineering.* 2013. Vol. 8, No. 6. P. 313–328. URL: <http://dx.doi.org/10.14257/ijmue.2013.8.6.31>

7. Federal'nyj zakon ot 02.05.2015 g. № 122-FZ «O vnesenii izmenenij v Trudovoj kodeks Rossijskoj Federatsii i stat'i 11 i 73 Federal'nogo zakona «Ob obrazovanii v Rossijskoj Federatsii». URL: <http://static.kremlin.ru/media/acts/files/0001201505020007.pdf>

8. *A. Broder.* On the resemblance and containment of documents // *Compression and Complexity of Sequences (SEQUENCES'97).* IEEE Computer Society, 1998. P. 21–29.

9. Spring Framework. URL: <https://projects.spring.io/spring-framework>

10. PostgreSQL: The World's Most Advanced Open Source Relational Database. URL: <https://www.postgresql.org/>

11. MyStem. Yandex URL: <https://tech.yandex.ru/mystem/>

12. Bullet Education Scheduling and Timetabling. URL: <https://www.bulletsolutions.com/education-scheduling-timetabling/>

13. Mimosa Scheduling Software. URL: <http://www.mimosasoftware.com/>

14. Snap Schedule. URL: <https://www.snapschedule.com/Industry/Education/>

15. Ofitsial'nyj sajt programmogo produkta BIT.VUZ.RASPISANIE. URL: <http://www.pulsar.ru/progs/1904/>

16. *Lim C., Sim E.* Production Planning in Manufacturing / Remanufacturing Environment using Genetic Algorithm // *GECCO 2005: Proc. of the 2005 Conference on Genetic and Evolutionary Computation,* 2005. P. 2217–2218. URL: <http://www.cs.bham.ac.uk/~wbl/biblio/gecco2005/docs/p2217.pdf>

17. Официальный сайт фреймворка Angular 5. URL: <https://angular.io/>

СВЕДЕНИЯ ОБ АВТОРАХ



АБРАМСКИЙ Михаил Михайлович – старший преподаватель кафедры программной инженерии Высшей школы информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета.

Mikhail Mikhailovich ABRAMSKIY – senior lecturer at Department of Software Engineering of Higher School of ITIS KFU
email: ma@it.kfu.ru



БАТЫРОВА Эльвира Феликсовна – бакалавр Высшей школы информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета.

Elvira Felixovna BATYROVA – bachelor of Higher School of ITIS KFU.
email: elviraa3805@gmail.com



МАРДАНОВА Айгуль Рустамовна – бакалавр Высшей школы информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета.

Aygul Rustamovna MARDANOVA – Bachelor of Higher School of ITIS KFU.
email: aigulmardanova96@gmail.com



АХМЕТЗЯНОВА Татьяна Алексеевна – магистр программной инженерии Высшей школы информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета.

Tatiana Alekseevna AKHMETZYANOVA – Master of Software Engineering of Higher School of ITIS KFU
email: tanya.push0494@gmail.com

Материал поступил в редакцию 31 июля 2018 года