

Во всех режимах реализована возможность выбора части света, реализован таймер и помощь в виде «+15 секунд».

В разделе «О приложении» представлены возможности приложения, название, актуальность продукта, средство разработки, автор и год создания (рисунок 6).



Рисунок 6 – О приложении

Литература

- 1 HTMLBOOK [Электронный ресурс]. – Режим доступа: <http://htmlbook.ru>. – Дата доступа: 27.02.2018.
- 2 Основы Vue.js [Электронный ресурс]. – Режим доступа: <https://metanit.com/web/vuejs/1.1.php>. – Дата доступа: 18.03.2018.
- 3 Vue.js – Прогрессивный JavaScript-фреймворк [Электронный ресурс]. – Режим доступа: <https://ru.vuejs.org>. – Дата доступа: 20.03.2018.
- 4 Макнейл, П. Web-дизайн. Идеи. Секреты. Советы / П. Макнейл. – СПб.: Питер, 2012. – 146 с.
- 5 Дунаев, В. Сценарии для Web-сайта. PHP и JavaScript / В. Дунаев. – СПб.: БХВ-Петербург, 2008. – 125 с.

УДК 519.95

Ю. А. Слепенко, А. Н. Осипенко, Н. Б. Осипенко

АВТОМАТИЗАЦИЯ ПЕРСОНИФИЦИРОВАННОЙ ОРИЕНТАЦИИ ПРИ ПОДБОРЕ ЧЕЛОВЕКОМ ВАРИАНТОВ ПОЛЕЗНОГО РЕШЕНИЯ

Разработанная версия клиент-серверного web-приложения обеспечивает поддержку принятия стратегических решений на примере профориентации за счет предоставления возможности пользователю прохождения психологического и антропометрического тестирования на клиентской части клиент-серверного приложения и обработки полученных данных на серверной части с целью формирования рекомендаций и индивидуального прогноза.

Каждый человек обладает набором уникальных характеристик и работа, простая для одного человека, может быть совершенно неподходящей для другого. Когда человек выполняет работу, неподходящую ему, он не только портит себе жизнь, он занимает место того, кому бы эта работа подошла больше. Для того чтобы ответить на вопрос: «Где человек может себя проявить наиболее эффективно, с самой высокой самоотдачей?» было решено создать программный продукт, подсказывающий человеку наиболее и наименее предпочтительные ему сферы деятельности и профессии. Для этого были решены следующие задачи: выбран инструмент решения поставленной задачи; спроектирована структура базы данных для покрытия всех возможных требований при выборе профессий и сфер деятельности для пользователя; разработан, протестирован и отлажен алгоритм выбора профессий и сфер деятельности; создана структура web-приложения; реализован алгоритм расчета простого, расширенного и средневзвешенного квадратов Пифагора; разработаны тесты для выявления предрасположенности человека к определенной деятельности или профессии; реализован инструмент психологического тестирования по серии тестов и обработка результатов психологического тестирования на клиентской части клиент-серверного приложения, а также аддитивная свертка результатов психологического тестирования; реализован алгоритм расчёта рекомендуемых сфер деятельности и профессий; готовый продукт развернут на внешнем сервере.

Разрабатываемое клиент-серверное web-приложение является продолжением реализации, описанной в [1], и предоставляет возможность пользователю прохождения серии психологических и антропометрических тестов с целью профориентации, а в перспективе – навигации индивидуальных решений в рамках таких направлений, как: подбор нужных людей или сложных интеллектуальных продуктов (например, информационных ресурсов, игр для детей и взрослых и др.), а также подбор видов деятельности (подбор профессии, видов отдыха) и т. п.

Формируемые web-приложением рекомендации основаны на алгоритме квадрата Пифагора и на оригинальных авторских разработках в области систематизации тем и сфер деятельности [1], а также на результатах психологического тестирования и их увязке с вышеупомянутой систематизацией. Сайт предлагает пользователю спектр альтернатив для выбора индивидуальных решений в рамках серии ранее упомянутых направлений. Таким образом, ответственность за выбор остается за человеком. Программа только позволяет, с одной стороны, сузить предоставляемое информационной средой огромное множество альтернатив и тем самым сэкономить время на принятие решения, и с другой стороны, –значительно снизить вероятность попадания в ловушку очередного модного веяния и выбора не лучшего или даже вредного решения.

Основным инструментом реализации явились: язык сценариев JavaScript, язык программирования PHP, фреймворк web-программирования общего назначения YiiFramework, фреймворк AngularJS, СУБД MySQL и архитектура построения web-сервисов типа REST. Пересечение таких технологий как Yii Framework и AngularJS позволяет следовать одновременно нескольким концепциям построения web-приложений. Широкие возможности Yii позволяют перенести логику роутинга по страницам на сам фреймворк, а так же избавляют разработчика от необходимости низкоуровневой работы с базой данных, вместо которой используется паттерн ORM (Object Relational Mapping). На сложных страницах, таких как страница психологического тестирования, либо страница регистрации, используется Angular-приложение, которое позволяет следовать паттерну SPA(Single-Page Application), в связи с чем уменьшается нагрузка на сервер, а пользователь может работать со страницей без ее перезагрузки. Для связи фронтэнда и бэкэнда приложения используются REST-сервисы. Схематичное описание работы приложения приведено на рисунке 1.

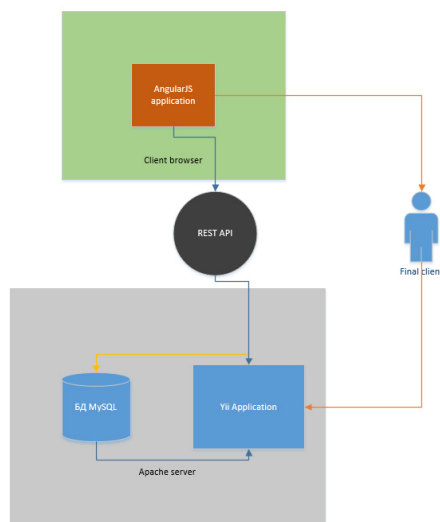


Рисунок 1 – Схема работы приложения

Разработанная база данных рассчитана на хранение больших объёмов данных, связанных с личностями пользователей, а также на удобный доступ разработчиков к этим данным для последующей обработки. Несмотря на то, что задачей разработки данной работы и соответствующего программного обеспечения является подбор подходящих человеку тем и сфер деятельности, а также профессий, в перспективе описываемое программное приложение ориентировано на решение упомянутого более широкого круга задач.

Каждая из перечисленных задач подразумевает использование своих собственных алгоритмов и некоторых уникальных результатов тестов. Хотя в целом структура базы данных для них похожа, была предусмотрена возможность добавления новых блоков в приложение. Видимо, подобную гибкость базе данных может дать отсутствие жесткой структуры для некоторых данных за счет использования сравнительно новой возможности MySQL – JSON-полей, на основании которых информация, хранимая в таких полях, будет иметь смысл только для специализированного средства ее обработки (модуля приложения). Примером такого поля является поле «Результаты» в таблице «Тестирование». Для дополнительного уточнения результатов тестирования пользователю предлагается сохранять информацию о людях, с которыми он состоит в родственных отношениях. Также для удобства пользователей предусмотрена возможность регистрации.

Структура базы данных разрабатываемого приложения должна включать в себя хранение всех необходимых для решения этих задач данных. Разработанная структура базы данных имеет вид, представленный на рисунке 2.

Таблицы базы данных, необходимые для непосредственной работы приложения в существующем виде, описаны в таблице 1.

Для работы с базой данных была использована предлагаемая по умолчанию в YiiFramework ORM-система: ActiveRecord, которая позволяет обращаться с таблицами базы данных, как с классами php, а с записями в них – как с экземплярами этих классов.

Основные алгоритмы работы приложения можно разделить на пять групп:

1. Расчет квадрата Пифагора для разных случаев (простой квадрат из 9-ти элементов, расширенный квадрат из 23-х элементов, средневзвешенный квадрат из 23-х элементов, рассчитываемый с учетом влияния родственников на этого человека.
2. Психологическое тестирование.
3. Аддитивная свертка результатов психологического тестирования.
4. Расчет подходящих сфер деятельности.
5. Расчет подходящих профессий.

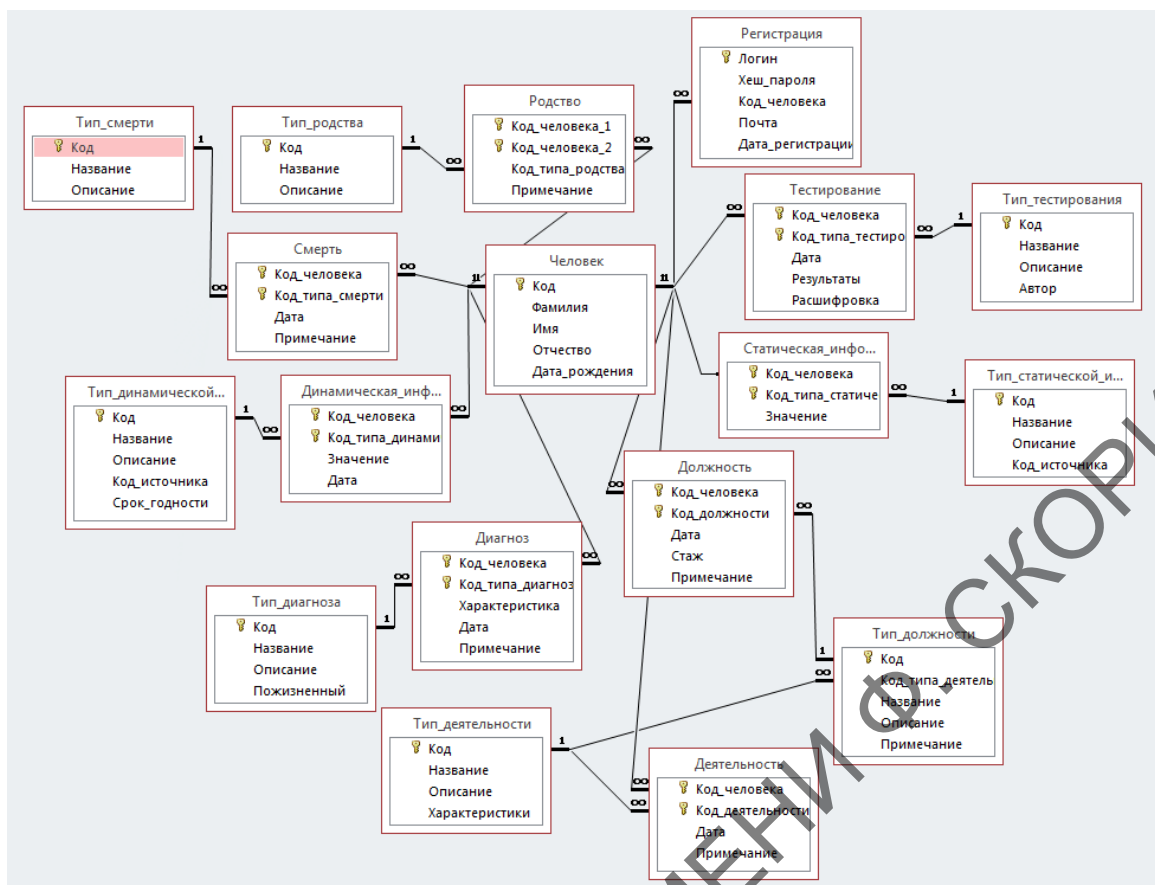


Рисунок 2 – Разработанная структура базы данных

Таблица 1 – Состав таблиц базы данных

Таблица	Назначение	Содержание
user	Базовая информация о пользователе	Идентификатор, ФИО, юзернейм, емейл, хеш пароля, дата рождения, роль пользователя в приложении
activity_type	Информация о сфере деятельности	Идентификатор, название, набор описывающих ячеек
user_relation	Описание родственной связи пользователя	Идентификатор, название, уровень близости
user_to_user	Родственная связь пользователя с другим пользователем	Идентификатор первого пользователя, идентификатор второго пользователя, идентификатор типа родственной связи.
user_to_activity	Описание связи пользователя со сферой деятельности	Идентификатор пользователя, идентификатор сферы деятельности, дата расчета, вес сферы деятельности
profession	Описание профессий	Идентификатор профессии, название, идентификатор связанной сферы деятельности, ячейки характеристик сферы деятельности, биты требований лидерских качеств
user_info	Связующая таблица для информации о датах рождения с привязкой по пользователям	Идентификатор, идентификатор пользователя, дата рождения

Таблица	Назначение	Содержание
Test	Описание инстанса психологического теста	Идентификатор, название, вес, комментарий, описание
question	Описание инстанса вопроса для психологического теста	Идентификатор, идентификатор теста, название, текст, url-адрес изображения
answer	Описание инстанса ответа на вопрос для психологического теста	Идентификатор, идентификатор вопроса, название, url-адрес изображения, текст, JSON-строка ячеек затронутых параметров личности, вес ответа
user_to_testing	Результат прохождения пользователем определенного психологического теста	Идентификатор, идентификатор теста, идентификатор пользователя, JSON-строка с выбранными вариантами ответа, JSON-строка с результатами тестирования после аддитивной свертки, дата прохождения тестирования
usr_to_profession_testing	Результат выполненного запроса на рекомендацию профессий с оценкой результатов	Идентификатор, идентификатор пользователя, количество пройденных тестов, общая оценка результатов тестирования, JSON-строка с результатами без влияния психологических тестов, JSON-строка с результатами после перерасчета влияния пройденных психологических тестов, дата прохождения тестирования

Алгоритмы решения этих задач были разбиты и перенесены в модели PythagorasSquare, UserToActivity, Profession.

Функции расчета простого и расширенного квадрата Пифагора принимают на вход дату рождения пользователя и выдают массив пар «ключ-значение» с элементами квадрата Пифагора. Логика расчета средневзвешенного квадрата Пифагора куда более сложная, и на вход функция принимает экземпляр класса User из базы данных. В случае отсутствия родственников, алгоритм выполнит расчет расширенного квадрата Пифагора для этого пользователя и вернет его (влияния родственников на человека нет, усложнение алгоритма не требуется). В противном случае функция найдет расширенный квадрат Пифагора для каждого родственника, и пересчитает основной массив базового человека с учетом весов соответствующих элементов массивов данных о родственниках.

Так же в классе PythagorasSquare находятся функции для расчета маски лидерских качеств пользователя, расчета ячеек характеристик пользователя, расчет уровней допустимости элементов квадрата Пифагора.

Функции расчета пользовательских сфер деятельности вынесены в класс UserToActivity, который изначально отвечал за интерфейс хранения экземпляров этого класса в базе данных. В классе находятся функции для полного и упрощенного алгоритмов расчета сферы деятельности и ее веса. Во всех функциях используются вызовы методов класса PythagorasSquare.

Функции расчета пользовательских профессий были вынесены в класс Profession, который изначально отвечал за интерфейс хранения экземпляров этого класса в базе данных. В классе находятся функции для полного и упрощенного алгоритмов расчета предпочтительной профессии и ее веса. Во всех функциях используются вызовы методов класса PythagorasSquare. Учет всех промежуточных результатов ведется в логах приложения.

Первая апробация предложенного подхода осуществлена на примере проблематики подбора профессий [1]. Исходный код продукта и скрипты создания базы данных располагаются в репозитории GitHub и доступны для просмотра и редактирования другим специалистам по адресу https://github.com/yslepianok/analysis_site. Программное решение развернуто на хостинге и доступно желающим в Интернете по ссылке <http://gsu-psychoanalysis.tk>.

Литература

1 Осипенко, А. Н. Автоматизация диагностики потенциальных качеств человека при выборе профессии / А. Н. Осипенко, Н. Б. Осипенко, Ю. А. Слепенко // Проблемы физики, математики и техники. – 2016. – №3 (28). – С. 88–96.

2 Слепенко, Ю. А. Структура базы данных для решения задач экспресс-подбора наиболее подходящих человеку профессий / Ю. А. Слепенко, А. Н. Осипенко, Н. Б. Осипенко // Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях: материалы XIX Республиканской научной конференции студентов и аспирантов, Гомель, 21–23 марта 2016 г.: в 2 ч. – Гомельский гос. ун-т им. Ф.Скорины; редкол.: О.М. Демиденко (гл. ред.) [и др.]. – Гомель, 2016. – Ч. 2. – С. 67–68.

УДК 004.93'1

И. В. Тимохин, Н. Б. Осипенко

О ВЛИЯНИИ РАЗМЕРА ОБУЧАЮЩЕЙ ВЫБОРКИ НА ТОЧНОСТЬ РАБОТЫ КЛАССИФИКАТОРА ВИОЛЫ – ДЖОНСА

В статье рассматривается влияние размеров обучающих выборок положительных и отрицательных образцов на точность получаемого классификатора Виолы – Джонса, который используется для обнаружения объектов. Точность получаемого классификатора сравнивается с полученными ранее результатами. Рассматривается, в каких случаях обучение классификатора unsuccessfully завершается досрочно.

Компьютеры уже давно используют технологии распознавания определённых черт на изображении. Однако результаты были далеки от успеха. Компьютерное зрение оказало на глубинное обучение невероятное влияние. Именно эти две техники в данный момент решают все задачи на распознавание. В частности, в распознавании лиц на фотографиях с помощью глубинного обучения преуспел Facebook. Это не простое улучшение технологии, а поворотный момент, изменяющий все более ранние представления: «Человек может с вероятностью в 97,53% определить, один ли человек представлен на двух разных фотографиях. Программа, разработанная командой Facebook, может делать это с вероятностью в 97,25% вне зависимости от освещения или того, смотрит ли человек прямо в камеру или повернут к ней боком» [1].

В [2] описывается система для распознавания людей по изображениям лиц. Описываемая система состоит из двух частей: первая часть выделяет имеющиеся на изображении лица, которые затем сравниваются между собой, во второй части происходит принятие решения о том, одинаковые или разные люди находятся на изображениях. В этой статье описываются попытки улучшения точности работы первой части, для чего обучались классификаторы на выборках различных размеров.

Для выделения лиц на изображениях в [2] используется реализация метода Виолы – Джонса из библиотеки OpenCV [3]. Для получения каскадного классификатора в [2] обучение производилось на выборке положительных образцов Georgia Tech face database [4], содержащей изображения 50 людей, снятых на протяжении двух или трёх сессий между 1 июня и 15 ноября 1999 года в Центре обработки сигналов и изображений в Институте технологий Джорджии (Center for Signal and Image Processing at Georgia Institute of Technology). Для каждого человека в этой выборке содержится 15 цветных JPEG изображения с разрешением 640 на 480 пикселей и средним размером лица на всех фотографиях около 150 на 150 пикселей. Кроме непосредственно